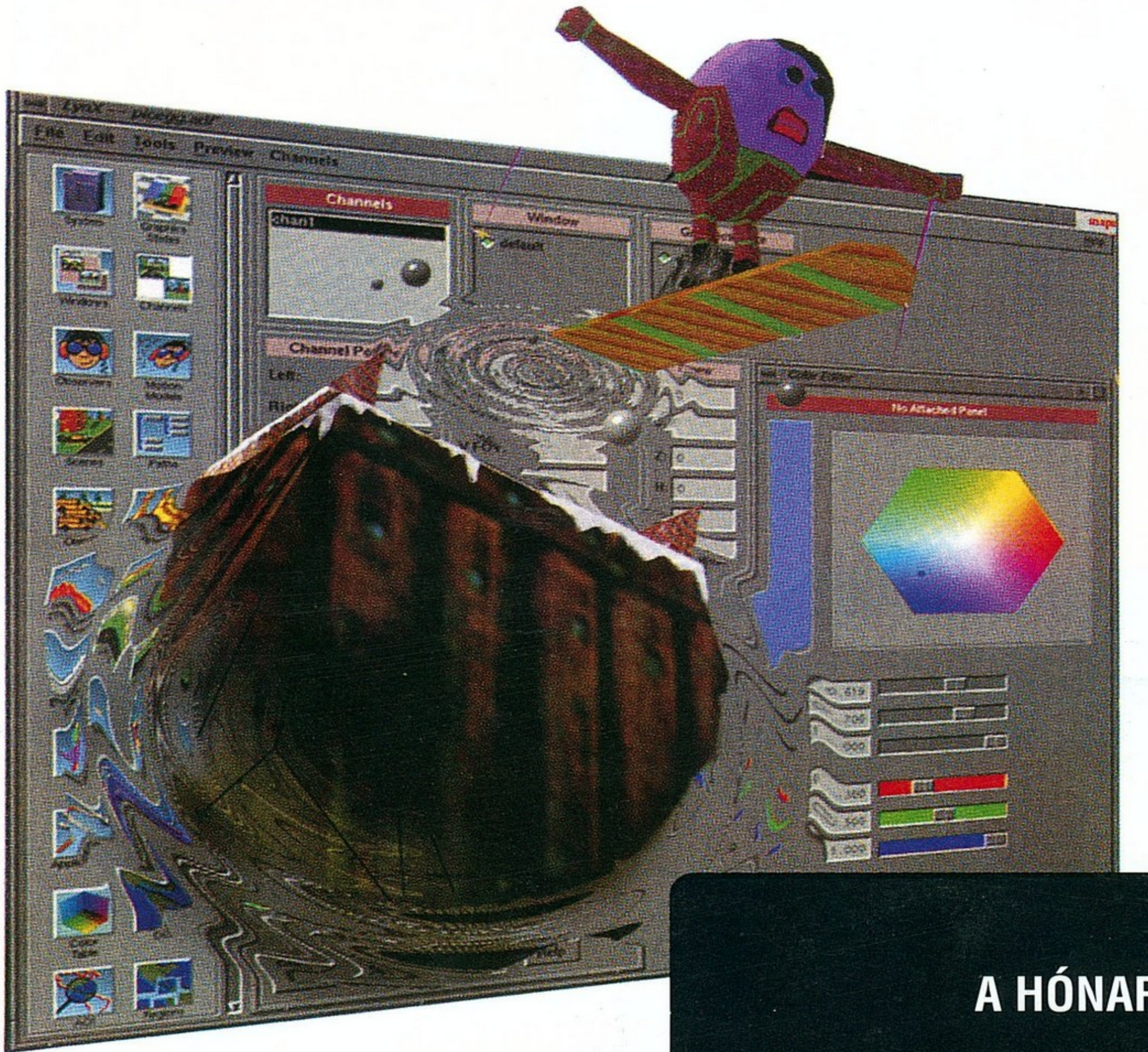
1995 / AUGUSZTUS ÁRA: 356 FT

ÚJ ALAPLAP

SZÁMÍTÁSTECHNIKAI MAGAZIN MÁGNESLEMEZ MELLÉKLETTEL



A LEMEZEN:

„Floppyátiratok"
Directory-kezelők
Egy szupertömörítő
Magyar editor
Kínai dáma

Elkékülni..., rajt!

A HÓNAP TÉMÁJA:

ABLAKADABRA

A sorsolás nyertesei

Elektronikus laboratórium PC-n

Hogyan jussunk A:-ról B:-re

Új IBM PC-korszak

Amikor a közepes az optimális

Crush — a tömörítők élősködője

ÚJ ALAPLAP

A Mikroszámítógép Magazin és az Alaplap hagyományait folytató számítástechnikai folyóirat

Megjelenik havonta, mágneslemez melléklettel

Főszerkesztő:
Faklen Pál

Főszerkesztő-helyettes:
Varga János

Szerkesztő:
Jakab Ágnes

A szerkesztőbizottság tagjai:
Barna László, Brüll Károly, Csórián Sándor, Feleki Zoltán, Herczeg József, Horlai János, Jánosi Tibor, Kis János, Nagy Gábor, Sík Zoltán, Szondi Egon János, Vargha Dénes, Vékony Tamás, Villányi László

Szerkesztőség és kiadó:
1538 Budapest I., Márvány u. 17.
Telefon: 156-3211 / 200, 214
Fax (manuális): 156-3211 / 201

Felelős kiadó:
Faklen Pál

Terjesztés:
Megyes Zsuzsa

Hirdetésszervezés:
Árvai Katalin, Bogácsi Mária, Pap Katalin, Tóth Zoltán

Külföldi hirdetések:
PubliCity
Reklám- és Médiaügynökség
1537 Budapest I., Márvány u. 17.
Telefon: 156-1182 Fax: 175-3539

Példányszámadatok hitelesítése:
Magyar Terjesztésellenőrző Szövetség
MATESZ
és Price Waterhouse

Nyomtatás:
Zalai Nyomda Rt, Zalaegerszeg
Felelős vezető:
Somogyi Tibor ügyvezető igazgató

Terjeszti:
A Magyar Posta Rt, a Nemzeti Hírlapkereskedelmi Rt, a Hírker Rt, a Kiadói Lapterjesztő Kft, számos számítástechnikai szaküzlet és más alternatív terjesztő

Előfizethető a kiadónál:
Új Alaplap Kiadói Kft,
1538 Budapest, Pf. 571
Átutalás: OTP 218-98017 / 501-017164-7
Példányonkénti ár: 356 Ft
Évi előfizetési díj: 3564 Ft

Külföldre terjeszti a Kultúra, H-1389 Budapest, Pf. 149

HU ISSN 1217-7598

A HÓNAP TÉMÁJA: ABLAKADABRA

(Összeállította: Faklen Pál)



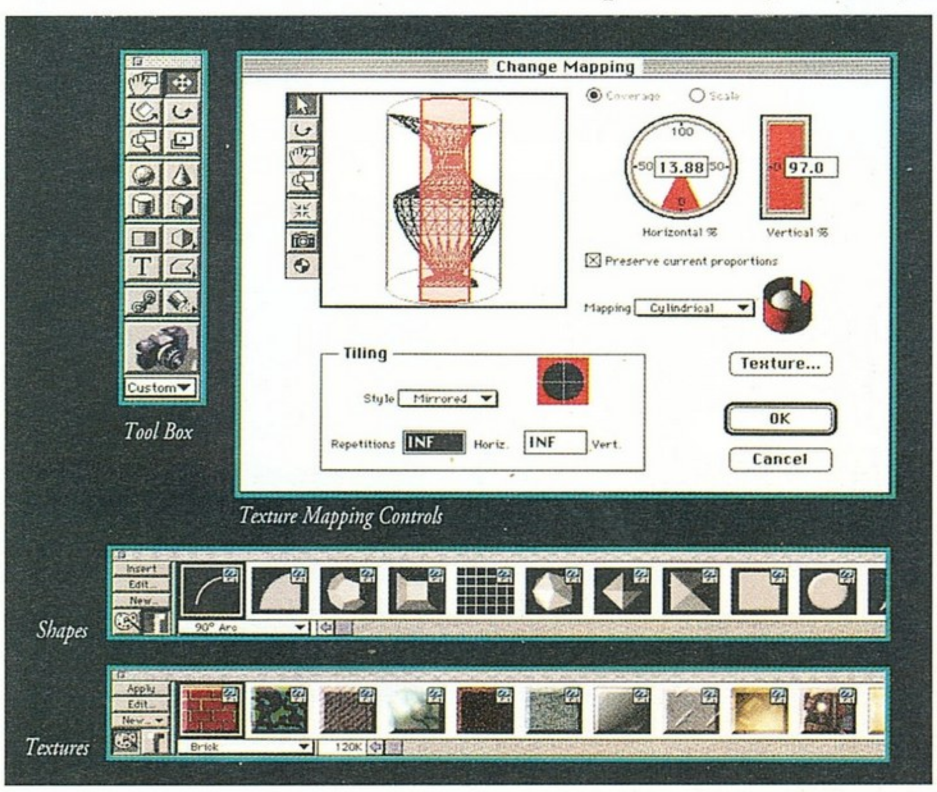
Van-e alternatívája?

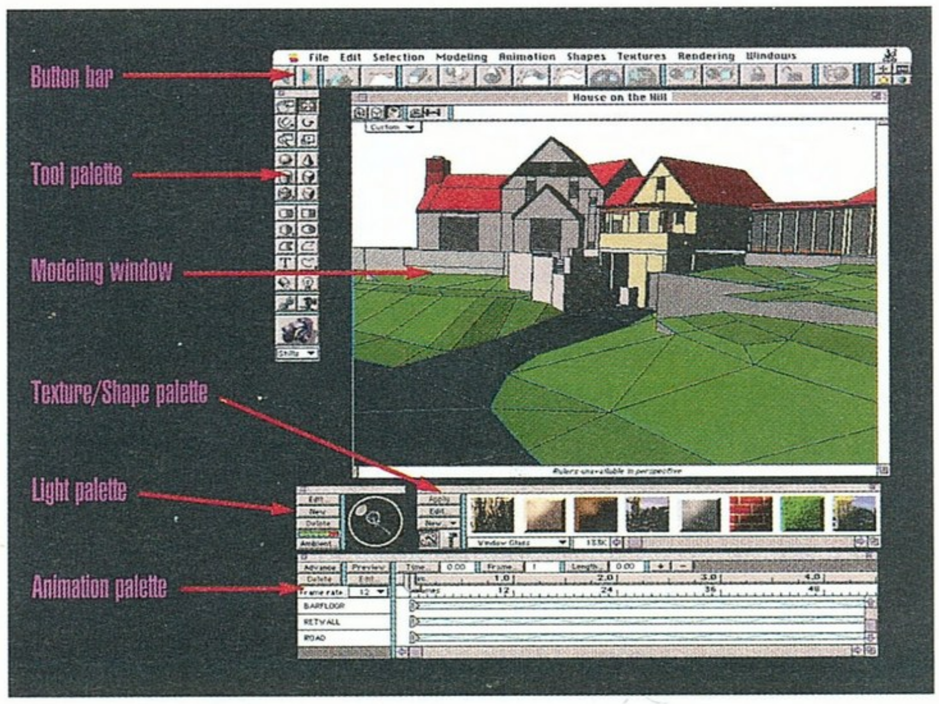
Mármint van-e alternatívája az MS Windows világának, tágabb értelemben a grafikus felhasználói felületnek, és mostantól kezdve az Oprendszer—Ablak házaspárnak. Van-e még választási lehetőségünk, vagy törődjünk bele abba, hogy kialakult egy számítástechnikai egypártrendszer? Akik esetleg már e kérdésfeltevésre is ingerülten reagálnak, mondván „Miért kellene, hogy a Windowsnak alternatívája legyen?", azokkal az itt következő oldalakon aligha fogunk szót érteni.

Különösen nehéz vitatkozni erről a témáról akkor, ha összekeveredik két fogalom: az *ablaktechnika* és az *Ablak*. Az egyik a számítógéppel folytatott „párbeszéd" fejlődésének szükségszerű állomása, amely a közvetlen kódolás, a parancsnyelv és a menürendszer után érkezett el és vált népszerűvé, valamikor pedig talán folytatódik a beszédfelismerés, a kézírásfelismerés, sőt egyesek szerint a gondolatolvasás gépi technológiájával. A másik ezzel szemben csak egy konkrét termék.

Az ablaktechnika és a grafikus mód közé sem lehet egyenlőségjelet tenni, de való igaz, hogy a grafikus kezelői felület bizonyos többletfunkcióival a karakteres működés nem ruházható fel. Nem is ebben jelentkeznek a szakmai nézeteltérések, hanem abban, hogy szükségszerű-e *minden* számítástechnikai feladatot grafikus felületre átvinni. S ha igen, akkor ennek egyetlen kvázi szabvány zárt világában kell-e megtörténnie, vagy inkább nyitott számítógépes környezetben? Ha pedig a racionalitás esetleg mégis a monolitikus megoldást diktálja, akkor miért éppen a Microsoft erőforrástékozló rendszerére kell mindent felépíteni?

E nehezen megválaszolható alapkérdések egyenes következménye, hogy technikai köntösben megjelentek a középkor teológiai hitvitái, meg az állampárti korszak „velünk vagy ellenünk" komplexusai. Amikor valamiről nem lehet teljes bizonyosságot szerezni, akkor a hit lép előtérbe. DOS-hívőkre és Windows-hívőkre oszlik a számítógépes szubkultúra, egy szakmailag erős és befolyásos *kisebbség* pedig keresi az alternatív „vallásokat". Mindhárom csoport közös jellemzője,





hogy hitük tárgya nem tölti el őket maradéktalan elégedettséggel. Nyíltan vagy titokban valami jobbat (is) szeretnének.

Bill Gates az amerikai sajtónak a Microsofttal szembeni barátságtalan hangvételére panaszkodik. És nem érti az okát. Való igaz, hogy más cégek is rendre megjelennek a piacon ESNS-szoftverekkel (ESNS = Elsőre Soha Nem Sikerül), velük szemben a sajtó mégis elnézőbb . Talán mert ott kisebb átmérőjű csapból folyik az, hogy *„mi vagyunk a legszebbek, a legjobbak, a legokosabbak, a legsikeresebbek..."*. Az MS-DOS és az MS Windows gyengeségeit még talán el is néznék, de a pénz és a hatalom arroganciájára a Föld túlsó felén is érzékenyek — ha nem is annyira, mint mi itt Európában. A „szabálytalan" piaci magatartásra viszont Amerikában reagálnak kifinomultabban, mint a vadkapitalizmusban tévelygő poszt-szocialista országokban.

Elvitathatatlanok viszont a Microsoft érdemei a számítástechnika *propagandájában* és *popularizálásában*. Ezt lapunk szerkesztési filozófiája is kezdettől fogva támogatja. A mindenki által kezelhető kommunikációs felület követelménye, a programok használatát megkönnyítő beidegződések kialakítása, az installálási rutinfeladatok automatikus elvégzése, a kompatibilitás és az átjárhatóság igénye... mind olyan vonások, amelyek szimpatikusak számunkra. (Annál nagyobb kér, hogy elég hosszú a beváltatlan ígéretek listája.)

Meggyőződésünk, hogy az MS Windows és a Microsoft hibáinak, illetve kifogásolható magatartásának ostorozása inkább válhat javára a számítástechnikának, sőt magának a Microsoftnak is, mint a tömjénfüst. Bár a grafikus felületnek egyelőre nincs alternatívája (még az Ablaknak is alig), igenis van alternatívája a gazdasági erőfölénnyel való visszaélésnek, a kidolgozatlanul piacra dobott termékek üzletpolitikájának... A Microsofton kívül sok múlik azonban az „alternatívokon", a versenytársakon, hogy a számítástechnikát (és az egész társadalmat) jobban szolgáló, hatékony, jövőbe mutató megoldások terjedjenek el. A versenyben mindig a *jobbiknak* kell(ene) győznie. Sajnos azonban (mint azt a kutyaközösségek szociológiai felméréseiből is tudjuk) általában nem a jobbik, hanem az *erősebbik* szokott győzni.

DOS vagy Windows

Ha választani lehet...

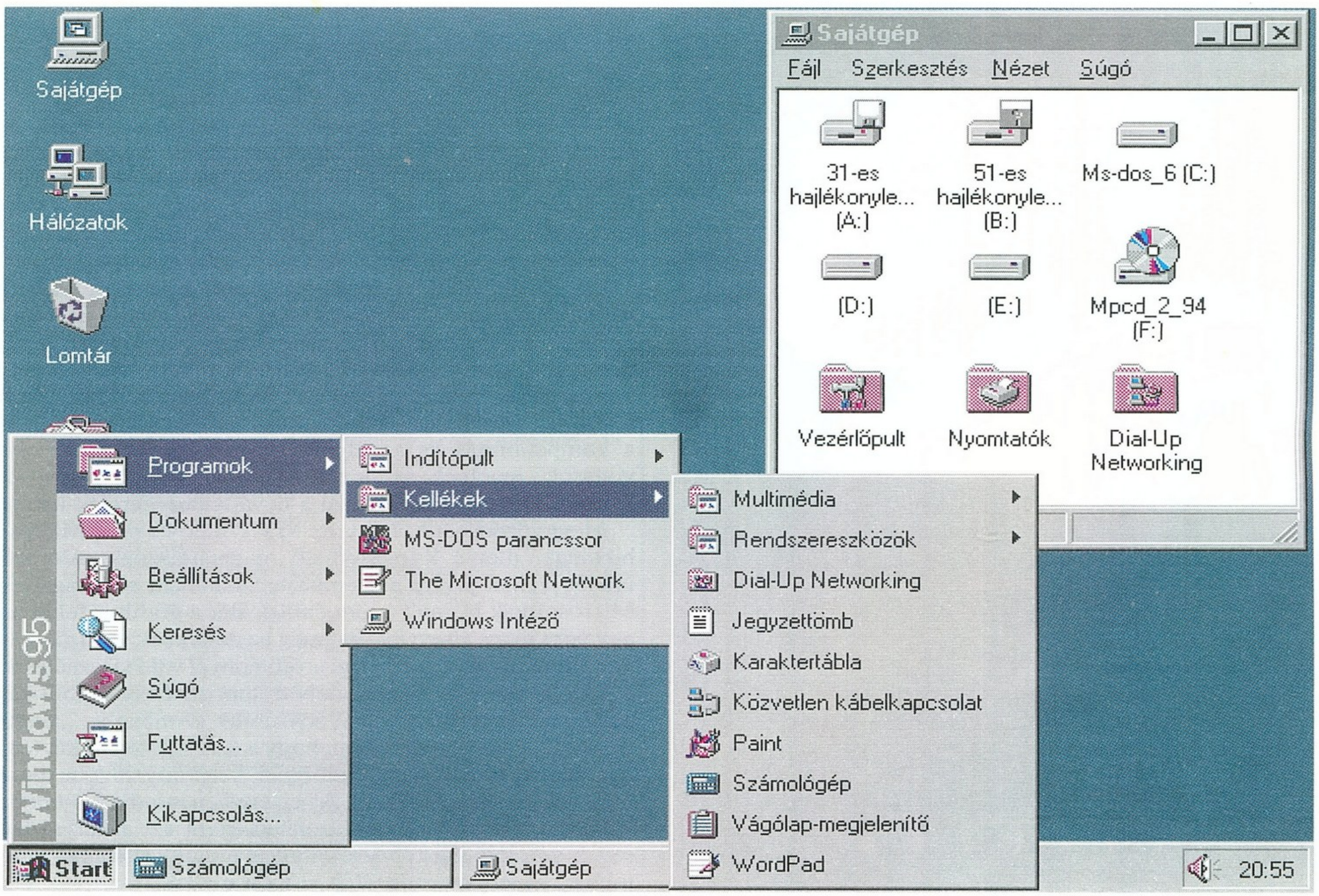
Amikor a Windows első változata megjelent, senki nem gondolta (talán maga Bill Gates sem), hogy abból minden idők legsikeresebb programja lehet, amely kikerülhetetlenné válik a számítástechnikát alkalmazók többsége számára.
A Windows 95 megjelenésekor érdemes most egy pillantást vetni az előd és a DOS-os környezet viszonyára, már csak azért is, mert ez az alternatíva sokaknak és sokáig még egyáltalán nem a múltat fogja jelenteni.

A Windows példátlan sikere azért is meglepő, mert ez ellen a Microsoft saját maga elég sok mindent megtett: az operációs rendszert és a grafikus kezelői felületet furcsa módon keverte, a programozási információkat visszatartotta, időnként megdöbbentően arrogáns és agresszív üzletpolitikát folytatott, a szoftverben súlyos tervezési hibákat hagyott, a kiérleletlenül forgalomba hozott program előnyeit aránytalanul felnagyította... A szakmai kommentárokban a Windows negatívumairól, megoldatlanságairól is esett ugyan szó, de többnyire megbocsátó, elnéző tálalásban, mintha e problémák csak egyedi jelenségek volnának, és nem a rendszer egészének lényeges gyenge pontjait mutatnák.

Uniformizált kezelői felület

A Microsoft szerint a felhasználó korábban elveszett a különböző alkalmazások sokféle kezelőrutinja között — hol F2-re nyílt meg egy fájl, hol Ctrl-O-ra, hol valami egészen másra. A Windows a kezelői felületet egységesítette, a műveletek zömének helyét vagy billentyűzetparancsát így csak egyszer kell megtanulni, s utána bármelyik új programban ugyanúgy használhatjuk azokat.

Nem vitatom, hogy ennek a szemléletnek, az egységesítésre való törekvésnek van alapja, nagyon sok ember számára a számítástechnika ezáltal valóban szimpatikusabbá vált. Magam részéről mégis az ellentáborhoz tartozom, mert a fenti előnyök részben igazak, részben nem, részben pedig egyáltalán nem olyan fontosak. Valóban, minden alkalmazásnál a bal felső sarokban ott van a File menü, s benne az obligát Open. Csakhogy mit kezdjünk az Open



File, Open Project, Open Library, Open Template esetekkel? Mégiscsak ismerni kell az adott alkalmazást, nem is beszélve a használat kevésbé triviális eseteiről. A komoly felhasználót nem zavarja, hogy különböző alkalmazásokban más-más billentyűk kellenek ugyanazon funkciók eléréséhez. De persze ez az egységesítés nem hátrány, csak nem akkora előny, mint hirdetik. (Gondoljunk a titkárnőre, aki csak egy programot, mondjuk egy szövegszerkesztőt használ egész nap.)

Segítség

Kevesen gondolnak rá, hogy a Windows tette szinte kötelező szabvánnyá a Help menüt, a vele járó „súgóállománnyal" és keresőrendszerrel együtt. Ma már valamire való alkalmazás nem készülhet enélkül, és valóban megkönnyíti a programok használatát. Persze a Helpnek is megvannak a vadhajtásai, sőt nemkívánatos következményei. Például, hogy akkor is elhanyagolják a kézikönyvet, ha arra bizony szükség lenne, vagy esetleg hogy több energiát fektetnek a Help csicsás kivitelezésébe, mint a programhibák kijavításába.

Ablak

Amiről a Windows a nevét kapta, az a konzekvens ablaktechnika. A méretezhető, odébb tolható ablakokkal készült MDI (multiple document) alkalmazások lehetővé teszik, hogy sokkal több információt tartsunk könnyen elérhető közelségben, mint amennyi a képernyőre férne, és mindig arra a részre tudjunk koncentrálni, amelyik éppen fontos, amelyikkel foglalkozni akarunk. Mindez a Microsoft kétségtelen érdeme.

Vágólap

Bevallom, hogy a clipboard (vágólap), amely arra szolgál, hogy egy alkalmazáson belül vagy alkalmazások között adatokat vigyünk át, kezdetben nekem nem tűnt fontosnak. Úgy gondoltam, hogy felesleges egy telefonszámot számtalan gomb nyomogatásával, hosszas egérmozgatással átvinni egy szövegszerkesztőbe, hiszen azt meg is lehet jegyezni, vagy netán cetlire felírni, majd direktben beírni. Feleslegesnek tartottam a rövid információknak ilyen átmeneti tárolót üzemeltetni. Hosszabb intenzív használat után ma már ezt másként látom. A táblázatban kiszámított értéket vagy pláne egy egész táblázatot tényleg érdemes így juttatni a helyére. Sok program (például egy kommunikációs szoftver outputja) érdekes lehet elmentésre és szerkesztésre, de nem várható el attól a programtól, hogy ehhez is adjon eszközöket. Ilyenkor a clipboard nélkülözhetetlen. Végül is a különböző alkalmazások közötti információcserének ez a statikus módja nagyon praktikus. Persze elvárnám legalább a hozzáadás és felülírás közötti választást, de még inkább a névvel ellátott vágólapot, hogy sok forrásból sokféle információt rakhassak sokfelé.

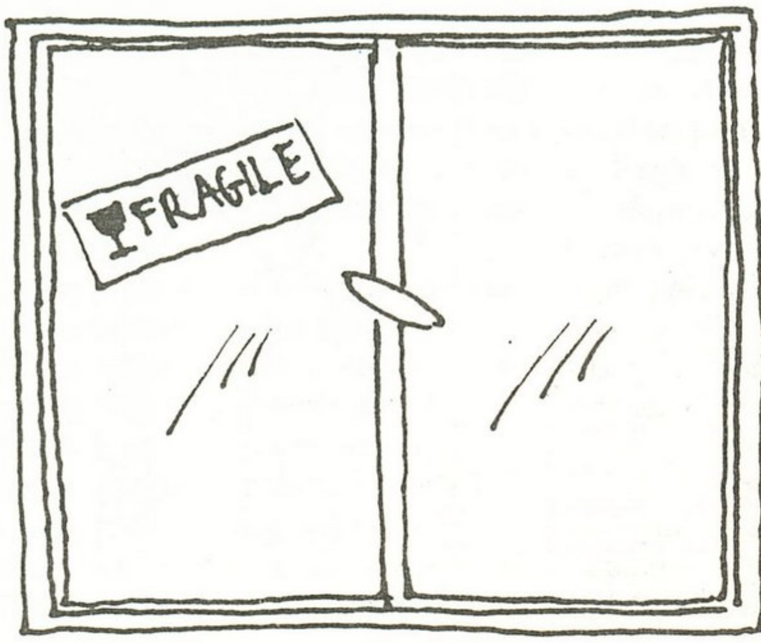
Multimicsoda

A vágólap átvezet a Windows másik, az operációs rendszer jellegével összefüggő aspektusához. Itt multitaszkos, azaz egyszerre több feladatot végrehajtó rendszert szoktak emlegetni. Ha a feladatok kicsik és nem foglalják le nagyon a CPU-t, akkor ez többé-kevésbé még igaz is. Érdekes látni, hogy miközben a szövegszerkesztővel dolgozunk, a háttérben futó kommunikációs program a részben elfedett ablak kilátszó részében csendesen írogatja a magáét. De mivel maga az operációs rendszer nem tudja magát megvédeni a garázda alkalmazásoktól, azaz a szabályokat nem tisztelő módon megírt programoktól, azok vidáman le tudják foglalni a gépet, ezért a Windows ebben félmegoldás. Amit kínál, az sokkal közelebb áll a „task-switching", a feladatváltás fogalmához, mint a valódi multitaszkhoz. Ráadásul még a rendes alkalmazások is olyanok, hogy ha sok erőforrást használnak, akkor a multitaszking csak látszat. Próbáljon valaki egy Excel tábla újraszámolása közben valami mást csinálni. Sajnos semmilyen mód nincs arra, hogy menet közben változtassuk meg a futó alkalmazások prioritását. Amit egyáltalán meg lehet adni, az csak egy meglehetősen gyengén dokumentált lehetőség az előtérben futó és a háttérben futó alkalmazások közötti időarány beállítására. Egyébként ez a multitaszking praktikus okokból is problémás. Ha valaminek az eredményére 10-20 mp-et kell várni, azt inkább megvárjuk, addig nem érdemes átkapcsolni valami másra. Ha hosszabb időt vesz igénybe a futás és elkezdünk mással foglalkozni, akkor pedig a háttérbe tett alkalmazás lassul le, így végül esetleg rosszabbul járunk, mintha kivárnánk a végét.

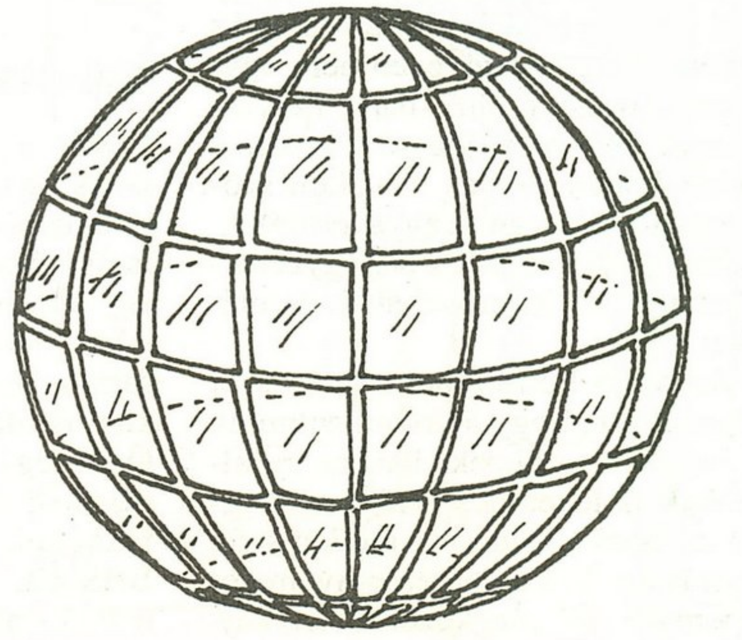
Forráshiány

Szintén tervezési probléma, hogy bizonyos kardinális fontosságú memóriarészek a 16 bites mód miatt csak 64 K méretűek. (Az ún. resource-okra gondolok). Ez a munkaterület a Windows írásakor bőven elégnek tűnt, de az újabb, cifrább alkalmazások felzabálják. Ha az MS Office két programja egyszerre fut, akkor mellettük mást már nem nagyon lehet csinálni. Sőt, ha ezt

WORD FOR WINDOWS



WORLD FOR WINDOWS



a két alkalmazást néhányszor becsukjuk és kinyitjuk, akkor látszik, hogy egyre kevesebb erőforrás (resource) áll rendelkezésre. Ugyanis még a Microsoft kirakati díszpinty-programjai sem képesek rendesen elengedni azt, amit egyszer lefoglaltak. Mit várjunk akkor egy egyszerű alkalmazói programtól?

Takarítás

Ha újabb programokat installálunk, akkor a Windows indulása alaposan lassul. Ugyanis minden program azt hiszi, hogy övé a világ, gátlástalanul telerakja a rendszert betűtípusokkal, eszközmeghajtókkal, segédprogramokkal. Ez még nem volna akkora baj, de utána szinte lehetetlen tőlük megszabadulni. A legokosabb uninstalláló program sem tökéletes, sőt, néha a szoftverhez adott, csak rá vonatkozó uninstall sem képes arra, hogy törölje önmagát a rendszerből. A DOS-ban általában a legsilányabb shareware program is nyom nélkül törölhető, legfeljebb az AUTOEXEC.BAT és a CONFIG.SYS korábbi változatát kell megőrizni, a Windowsban viszont minden mindennel összefügg, és a dokumentáció részleges volta, meg a számtalan változatban keringő .DLL-ek miatt nagyon nehéz egy jól működő korábbi állapotot visszaállítani. Így azután egyre telik a merevlemez és a memória. Vagyis nagyobb winchestert és több memóriát kell venni, bár ez utóbbi — visszagondolva a resource-ról mondottakra — nem sokat segít. Még a leggyorsabb processzor mellett is meggondolandó tehát, hogy milyen alkalmazások azok, amelyeknek a windowsos változatát érdemes használni és mikor jobb a „bevált" DOS mellett maradni.

Szövegszerkesztés

Paradox módon a **számító**gépet leggyakrabban **szöveg**szerkesztésre használják. Ez még inkább igaz a Windows esetében, főleg ha szövegen nem egyszerűen szöveget értünk, hanem olyasmit, amit ma már joggal neveznek inkább dokumentumnak. Ez a fajta szöveg ugyanis tele van beágyazott táblázatokkal, grafikonokkal, lábjegyzetekkel, betűtípusok kavalkádjával, emblémákkal.

A fanyalgó konzervatívoknak is el kell ismerniük, hogy a dokumentumok ettől nemcsak szebbek, hanem tartalmasabbak is lehetnek. Ami a szövegszerkesztésnél a Windows mellett szól, az nem is annyira a végleges nyomatot a képernyőn jól megközelítő látvány, hanem inkább a többi alkalmazással való adatcsere lehetősége, beleértve a már említett beágyazáson felül elsősorban a kényelmes, közvetlen faxolhatóságot.

Azért azt érdemes megjegyezni, hogy egy 50-100 oldalas dokumentum behívása már jelentős időt vesz igénybe, hiába van gépünkben gyors processzor és akár 8 MB memória. Ha pedig másik formátumból konvertálunk, akkor bizony perces várakozásokra kell számítani. (Nem szólva az eredmény kétes voltáról.)

Számolótábla

A számolótáblák eredeti funkciója remekül ellátható DOS-os programokkal is, de a grafikák beillesztése és más alkalmazásokkal való együttműködés miatt ezek is egyre gyakrabban megcélozzák a Windows-felületet. Nekem ugyan vadhajtásnak tűnik a cellaszintű formázás betűtípusokkal és egyéb kiemelési eszközökkel, de meg kell hagyni, hogy az eredmény sokszor lenyűgöző — még ha olykor egy kicsit sokat is kell rá várni.

Adatbázis

Talán legkevésbé való Windows alá az adatbáziskezelés. A külcsín itt is gyönyörű, de a tartalom, a teljesítmény érthetetlen módon siralmas. Már egy 3-400 KB-os adatbázisállomány feldolgozása is meghökkentően időigényes, saját nem kalibrált méréseim szerint akár tízszerese is lehet egy rendes DOS-os alkalmazásénak. Sokszor jobban megéri egy összesítő táblát ott elkészíteni és csak az eredményt átemelni Windows alá. Ennek a lassúságnak tényleg nem találom a magyarázatát. A grafikus felület és a multitaszk persze hozzáad valamit a futási időhöz, de hogy ennyit...? Őrizkedjünk a nagy fájlok Windows alatti feldolgozásától!

Kommunikáció

Két olyan terület van, ahol a Windows igazán jeleskedik. Az egyik a kommunikációs programok használata, melyekhez normál körülmények között egy átlagos 286-os gép is elegendő, tehát Windows alatt is elég szépen futnak, méghozzá kényelmes és praktikus módon rendszerint a háttérben. Öröm egy szöveget tovább szerkeszteni, miközben egy másik éppen békésen faxolódik; vagy letölteni egy fájlt egy BBS-ről, miközben valami egészen mást csinálunk. Ha azonban 9600 bps fölötti sebességgel akarunk dolgozni, akkor nem árt hozzá valami jobb drivert beszerezni.

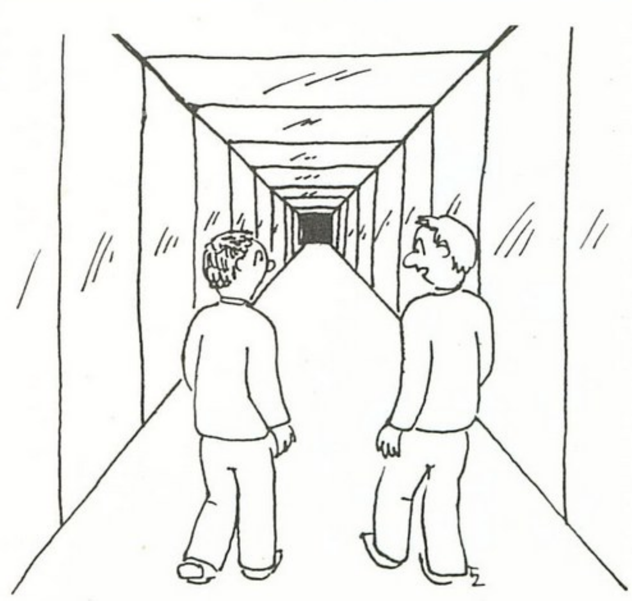
Grafika

A grafikus felülethez leginkább illeszkedő másik feladatcsoport érthető módon a grafikus alkalmazások nagy családjáé. Legyen szó egyszerű rajzolásról, DTP-ről vagy CAD programról, festőprogramokról vagy grafikonkészítésről, fotóretusálásról vagy folyamatábráról... mindenhez van sokféle alkalmazás, és bár ezek nagyon sokat tudnak, rendszerint itt is lassúsággal kell fizetni a teljesítményért. Ma a PC a grafikus alkalmazások jelentős hányadában egyenrangú versenytársa lett a Macintoshnak. Persze ezek az alkalmazások igénylik a legtöbb memóriát, a legnagyobb lemezfelületet és a leggyorsabb videokártyát.

QED (quod erat demonstrandum)

Sokan szeretik a Windowst, és használják nyakra-főre. Mások annyira „utálják", hogy akkor sem nyúlnak hozzá, ha megérné. Egyik szélsőségnek sincs igaza. Ha nem érzelmi kérdésként kezeljük, csupán a praktikus szempontokra figyelünk, akkor a Windows nagyon hasznos lehet olyankor, amikor a „testére szabott" alkalmazást kell futtatni. Olyan feladatoknál viszont, amelyekhez nem a Windows grafikus felületre történő rátelepedés az igazi megoldás, kerülni kell a használatát. Amikor a különböző alkalmazói felületeken közel azonos értékű megoldások kínálkoznak, döntsön mindenki a saját ízlése szerint. A pluralista, demokratikus szisztémáknál jobbat eddig egyetlen társadalom sem produkált — tanulta valamikor Bill Gates is az iskolában.

Horlai János



— Én már látom a Windows-alagút végét!

A Windows karrierjének titka

Miből (és hogyan) lesz a cserebogár?

Mi a Windows? Felhasználóbarát tömegtermék? „One billion dollar Bill", azaz Amerika leggazdagabb emberének szerencsekovácsa? A Macintosh grafikus operációs rendszerének koppintása? Világszabvány? RAM-faló faló, amelyet Gates ügyes reklámfogással minden PC-be beleálmodik, és utána az álom valósággá válik? (A programok megbízhatósága pedig álommá?) Kábítószer, amely szenvedélybetegséghez vezet? Egy megbízhatatlan hardveren egy még megbízhatatlanabb szoftver? Keressük a választ.

Microsoft at Work, Microsoft at Home. Nincs már messze az az idő, amikor borotválkozó tükör helyett a képernyőre meredve egérrel manőverezünk a párhuzamos portra kötött villanyborotvánkkal és jóleső áhítattal vesszük tudomásul, hogy Gates immár nemcsak a zsebünket vágja meg. But this has to be the Final Cut. És felébredünk. De akkor már túl késő...

A nagy ötlet

Hogyan lett a Windows a levitézlett DOS verziók tetemein egyre feljebb kapaszkodva a világ legnépszerűbb grafikus kezelői felülete?

A Microsoftot a kezdetek kezdete óta izgatta az Apple személyi számítógépeit méltán sikeressé tevő grafikus operációs rendszer. Még 1984 táján maguk is megírták a maguk klónját. Talán Windows 1.0-nak hívták. Otromba volt. Amilyen pici, olyan lassú. Megfelelt egy parancskiadásra alkalmas shellnek, saját alkalmazás a kötelező analóg és digitális órán kívül csak egy primitív szövegszerkesztő volt, fix betűkkel. Csak a legelvetemültebb szenvedélyes programguberálók találtak rá akkoriban. Senki sem vette komolyan. Talán csak Gates.

Mintegy két évvel később kihozta a még mindig monokróm 2.0-ás változatot. Ehhez néhány elhivatott programozó már gyártott egyet s mást. Például fonteditort, a 2.1 verziót megélő DOS-os Excel ultrakényelmetlen változatát, és a labilis Word for Windows 1.0-t. Különösebb nemzetközi érdeklődés nélkül kevés különc önsanyargatása jelemezte a Windows akkori felhasználótáborát.

Ekkor még a Microsoft vállvetve dolgozott az IBM-mel a megváltónak kikiáltott OS/2-n, ahol is a kitűzött cél egy valódi multitaszkos operációs rendszer lett volna.

Akkoriban több OS/2 Presentation Manager Development Kit fogyott, mint Windows Software Development Kit (SDK). Ráadásul mindkét termék indokolatlanul drága és tartalmát tekintve meglehetősen hézagos volt. A közös gyermek már vagy két éve nem akart megszületni, ami szakításhoz, sőt nyílt ellenségeskedéshez vezetett. Gates veszni hagyta a közösen fejlesztett kódot, amely 80-90 százalékban a Microsofté volt, és az XT-n is elinduló Windows 2.1-es változatot összevonva a 286-oson futó Windows változattal, kihozta a Windows 3.0-s verzióját. Ekkor többen is felkapták a fejüket, és úgy gondolták, hogy ez a már színesben pompázó program az OS/2 kiütésére készült. Ami igaz is volt, pedig talán maguk a programírók sem vették elég komolyan.

A Microsoft-csapatot akkoriban már a PC-s világ legjobb szoftveresei közé sorolták, de a szoros élmezőnyben legfeljebb fej-fej mellett haladhattak a Borland, az IBM, a Compaq, a Lotus, az Ashton-Tate, a Novell, a Digital Reserach, a WordPerfect gárdájával. Az igazi áttörést árstratégiájukkal érték el, feltehetően Steve Ballmernek, a Microsoft üzlethez is értő vezéregyéniségének ötlete alapján. Divatba jött ugyanis a nagy hardvergyártókkal megállapodni, hogy minden eladott PC-hez adjanak operációs rendszert, sőt más fontos programokat is. Erre a hardverbeszállítások mintájára OEM-szerződéseket kötöttek. (OEM = Original Equipment Manufacturer = összeszerelés eredeti alkatrészekből.) Miközben azonban a vetélytársak 40-50 dollár alatt nem kótyavetyélték portékájukat, a Microsoft a mindenkori DOS és Windows árát drasztikusan csökkentve 10 dollár alatti összegért sózta rá azokat a hardvergyártókra. (Az IBM a PC DOS-t semmi pénzért nem adta, az OS/2 nem volt olyan állapotban, hogy odaadható lett



volna, a Digital Research DR DOS-a pedig magasabb ára miatt nem nagyon terjedt el.) Gates számítása bevált. A folyamat öngerjesztővé vált. Az új gépek mellé egyre több gyártó akarta az MS-DOS-t és a Windowst adni. Az pedig egyenesen a hardvergyártók kezére (és zsebére) játszott, hogy a windowsos programok futtatásához egyre több RAM, egyre gyorsabb processzor, egyre nagyobb kapacitású merevlemez kellett, és a „szinte ajándékba adott" OEM szoftver egyre kelendőbb lett.

A PC-s világ adóbehajtója

A hardveres OEM-koreográfiához hasonlóan hálózta be később a Microsoft a szoftverfejlesztőket. Ellentétben a többi nagy szoftvercéggel, a Microsoft bőkezűen osztogatta a Windows fejlesztőeszközeit, így hamarosan több-száz programozó csapat érezte fontosnak, hogy alkalmazásokat gyártson a Windowshoz.

A felhasználó pedig egyszer csak azt vette észre, hogy felbillent a szoftverpiac addig megszokott egyensúlya (szövegszerkesztő = WordPerfect; táblázatkezelő = Lotus 1-2-3; rajzolóprogramok = a Digital Research GEM rendszerén alapuló programok; adatbáziskezelés = dBASE, esetleg kis részben Clipper.) Üstökösszerűen tűntek fel új cégek a Windows oldalvizén evezve: a Corel Systems CorelDraw-ja, a Micrografx Designere, Picture Publishere, nem beszélve az egyre jobb Macintosh átiratokról (Fractal Design Painter, PageMaker, QuarkXpress, Freehand, Adobe Illustrator, Adobe PhotoShop stb.)

Persze a Microsoft programozói és ügyvédjei sem tétlenkedtek. Nyíltan monopóliumra törekedtek az általuk későn és pongyolán közzétett szoftverszabványokkal (OLE, OLE2, ODBC stb.), valamint ügyes bevásárlásaikkal. Ezekre néhány jellemző példa: 1987-ben az eredetileg Mac-re írt PowerPoint megvásárlása a ForeThoughttól, 12 millió dollár készpénzért. 1991-ban az angol Dorling Kindersleytől vett Multimedia Beethoven lett a Microsoft Home sorozat alapköve. Ugyanebben az évben az akkor még nem létező Access legnagyobb vetélytársának, a FoxPrónak a bekebelezése. Ez a harácsolási láz napjaikban is megállíthatatlan lendülettel tart, bár néha az USA versenyhivatala jó ütemben („Intuitív" módon) közbenyúl. Hát így megy ez.

Azóta a Windows révén a Microsoft a PC-s világ legnagyobb és legügyesebb adóbehajtója. A fél világot belekergette az Office nevű snóblijátékba. (Milyen jól jár a felhasználó, hiszen egyenként 500 dolláros három szoftvert plusz egy 700 dolláros Mail szervert igénylő ingyenes Mail licencet tartalmazó dobozt ad 6-700 dollárért). Az átlagfelhasználó a csomag összes szolgáltatásának jó, ha 10 százalékát kihasználná, de nem teheti, mert mindig rohannia kell az aktuális frissítésért (az Office három komponensének új változatai természetesen időben szétszórva jelennek meg). Ez a mézesmadzag annyira jóízűnek találtatott a mameluk felhasználók körében, hogy a Microsoft rájött, ha egy vállalatnál N darab PC van, legyen a szorzó ezentúl N. Tiszta haszon, az MLP, MOLP és egyéb furfangos vevőkopasztó licencvásárlási konstrukcióknál már a doboz, a kézikönyv és az adathordozó média is megspórolható. És a sok sznob dalolva fizet. Ha nem, akkor megsuhogtatják a BSA vagy némely országokban az SPA nemlétező kardját (hazánkban 5 év börtönbüntetéssel fenyeget a mumus). De felhasználói érdekképviselettel valahogy senki sem foglalkozik. Pedig a vevőnek a problémás esetek többségében igaza van. A szoftver-licencszerződések a gyártót minden felelősség alól mentesítik (maximum adathordozót cserélnek), ha bármi nem OK, a vásárló a hunyó. Vajon csuklik-e néha Gates, aki annak idején privát fejlesztéseire lopta a gépidőt tizenéves korában a seattle-i egyetemen?!

Poloskatenyésztés nagyban

Miért van az, hogy senki nem csinál Windows alá egy egyszerű, gyors, salangmentes szövegszerkesztőt, amelynek nem is kellene többet tudnia, mint a szokványos írógépfunkciók és a WYSIWYG (Print Preview) mellett néhány egyszerű blokkművelet elvégzését, és esetleg egy univerzális helyesírás-ellenőrzést. Slussz. És ezt lehetne adni 50 dollárért (azaz 10 ezer forint alatt). A Microsoftnak és a többi nagy szoftveróriásnak nincs ilyen terméke. (Vigyázat, a Works nem ilyen! akár kell, akár nem, abban ott van adatbáziskezelő, táblázatkezelő, grafikonrajzoló, egy rakás varázsló stb.) De egyszerűségről szó sem lehet, mert a fő marketingcél: túlzott igényeket támasztani és azt még eltúlzottabban kielégíteni. Ebben a nagy túlszolgáltatásban aztán már csak a legdörzsöltebb Windows-felhasználóknak tűnik fel az, hogy amiért egy új csomagot agyonreklámoztak, az az opció köhög a legjobban. Mert még nincs kész. De sebaj, majd jobb lesz az N.M változatban. Amiről csak akkor tudunk meggyőződni, ha már megvettük. Mi is van a pénzvisszaadási garanciával?!

A számítógép (elvben) arra való, hogy speciális műveletsorokat szolgai módon minden esetben determinisztikusan, azonos módon hajtson végre, ha lehet a szélnél is sebesebben. Zárja ki, javítsa ki a tökéletlen ember hibázásait. Amikor töltőtollal ír az ember, a tintát elcsöppentheti, elmaszatolhatja. Aki ad valamit az esztétikumra, az ilyenkor az adott lapot elkezdheti újra. De mi van például a Word for Windowsban? Az írást nem lehet elkenni, nem tudunk pacát ejteni. Hurrá! De leselkedik ránk néhány, a programozó(k) által előre nem látható, ún. fatális hiba (hogy a fenébe is van ez?), úgymint: General Protection Falióra, Insufficient Memory, Disk Full stb... és adataink egy része huss, elszáll. Hát nem ott vagyunk, ahol a part szakad?

Nézzünk mélyen saját szemünkbe a Windows villódzó képernyőjének visszatükröződése nyomán: az így tömegesen vásárolt szoftver valóban tényleges termelőerőnek vehető? Hasznot hajt? (A Microsoftnak igen.)

Mondják, hogy a Windows története hazugságok (fogalmazzunk finomabban: be nem váltott ígéretek) története. Többfeladatos futtatás. Skálázható vektorfontok használata. Adott nyelv szerinti karakteres rendezés. Plug and play. Koprocesszor támogatása több taszk esetén. Ezek többségére a Windows 95 ad majd csattanós választ, ami négy szóban összefoglalva az eddigi jelek szerint a következő lesz: „MÉG MINDIG NEM TELJESEN." De akkor is ez a jövő útja — mondja Bill Gates.

Herczeg József



— Mi még használjuk a GEM-rendszert.

Közlekedés az egyirányú utcában

A piac az úr, nem a technológia

Az Új Alaplap ritkán szokott interjút közölni. Nem azért, mintha ez a műfaj idegen lenne lapunktól, hanem mert a jegyzetelt vagy magnóra felvett beszélgetésekből a nyomtatott szöveg kicsiszolása több munkát követel, mint megszerkeszteni egy kapott cikket, vagy különböző forrásokból közvetlenül megírni azt. És hát szerkesztőségünk igen kislétszámú csapat... Időnként mégis rászánjuk magunkat, ahogy most is. Köszönjük, hogy Drajkó László, a Novell kereskedelmi igazgatója vállalta a válaszadást szakmai faggatódzásunkra.

● Úgy tűnik, hogy a személyi számítógépeknél a grafikus kezelői felületek elterjedése szükségszerűen következett be. Volt rá igény, volt hozzá technikai lehetőség... De vajon miért éppen az MS Windows lett a „befutó"? Miért nem egy másik, például a GEM, amely helytakarékos, gyors, hatékony, kiforrott rendszer volt, már az MS Windows felbukkanása előtt.

— Én azt hiszem, hogy ennek fő oka egy üzletpolitikai döntés volt, amely hosszú távra kihatott. Annak idején az IBM a Microsoftot választotta a PC-s operációs rendszer szállítójának, nem pedig a lehetőségként ugyancsak komolyan felmerült Digital Research céget. (Az anekdota szerint egyébként egészen apró kis szervezési bakinak köszönhető, hogy mindez így alakult.)

A PC-s világ tehát behajtott a Microsoft egyirányú utcájába, s onnan a mai napig sem találta meg a kiutat. A kvázi ipari szabvánnyá vált PC-kre a Microsoft lett az operációs rendszer „udvari szállítója", annak ellenére, hogy mások hamarabb jelentek meg, és szakmailag egyértelműen jobb alternatívákat kínáltak.

Amint a Microsoft megerősödött az operációs rendszerek piacán, olyan szerződéseket kötött minden PC-gyártóval, hogy gyakorlatilag csak az ő szoftvere élt. Ahol valaki monopóliumot kap, ott azt csinál, amit akar. A Microsoft DOS-nak ezért nem lehetett igazi alternatívája egyebek közt az IBM PC DOS vagy a Novell DOS, noha szakmailag mind a kettő jobban működő, fejlettebb eszköz volt.

A grafikus felhasználói felülettel kapcsolatban azután a világ hasonló helyzetbe került. A Microsoft saját DOS-ára épülő grafikus felületet csinált. S bár az egyszerűség kedvéért sokan az MS Windowst operációs rendszernek titulálják, az valójában nem az, hanem egy operációs rendszerre rátelepített, a Windows 95 esetében pedig már egyenesen azzal összenövesztett grafikus felület. Szintén elterjedt az a téves szóhasználat, hogy egyszerűen csak Windowst, windowsos felületet, windowsos programot emlegetnek, olyankor is, amikor valójában MS Windows, MS windowsos lenne a korrekt kifejezés. A Windows nem termék, nem márka, hanem egy felületkezelési technológia, nem a Microsoft tulajdona. Más cégek is használják (lásd X Windows), és bárkinek jogában áll önálló ablaktechnikát kidolgozni.

Az eddigiekből persze az is következik, hogy az MS Windows versenyképes alternatívája csak egy operációs rendszer és a hozzá illesztett grafikus felület együttes fejlesztése lehetett volna. A Digital Research ehhez kicsi volt mint cég. Hiába állt szakmailag magas színvonalon, nem volt elegendő pénzügyi alapja, hogy befuttassa a GEM grafikus felületet.

Ráadásul az alkalmazói programok előtérbe kerülése megnehezítette a nagyobb váltásokat. A felhasználónak nem az operációs rendszer a fontos, hanem az alkalmazás. És az alkalmazói szoftverfejlesztők olyasmibe hajlandók munkát fektetni, ami piacot, forgalmat ígér. Az MS Windowst helyettesítő alternatív megoldás realitásának tehát nem szakmai, hanem üzleti akadályai voltak.

● Most a Windows 95-tel létrejön egy még szorosabb házasság az operációs rendszer és a grafikus felhasználói felület között. Ez persze nem valami nagy újdonság, hiszen a Macintosh 10 évvel korábban eleve így indult. Vajon a jövő is ilyen szimbiózisok irányba mutat, vagy olyan nyitott, rugalmas operációs rendszerek felé, amelyekre bármilyen grafikus felhasználói felület könnyen ráültethető?

— Azt hiszem, hogy e téren alternatívák mindig lesznek, és inkább a kisebb piaci részaránnyal rendelkezők lesznek kénytelenek alkalmazkodni a vezető helyzetben lévőkhöz. Jelen pillanatban például piacon van a Windows 95-nél jóval hamarabb bevezetett, fejlettebb, jobb grafikus felület, az IBM OS/2. Mégis az várható, hogy alternatívaként az OS/2-nek a Windows 95-ös támogatása biztosan meg fog jelenni, de fordítva a Microsoft aligha tesz ilyet. Legfeljebb egy Microsofttól független cég hoz majd ki olyan programot, amely támogatja az OS/2-re írt programok futtatását MS Windows alatt.

A jelenlegi helyzeten csak a piac tudna változtatni, de a fejlesztők ezt nem nagyon akarják, mert az ő érdekük elsősorban az egységesség. A Microsoft nem azért lett piacvezető ezen a téren, mert terméke annyira jó volt, hanem azért, mert valami egységeset szállított, egységes hardverplatformra.

● Ha a Windows 95 eddigi kritikáit nézzük, az derül ki, hogy a szakma nem igazán fogadja lelkesedéssel. Szedett-vedettnek, látszatmegoldásnak, kvázi előrelépésnek tartják. Ennek ellenére, milyenek az új MS Windows elterjedésének esélyei?

— A szakemberek zömének véleménye valóban elég pejoratív. Más kérdés, hogy a Microsoft milyen mértékben, milyen ütemben és milyen módszerekkel tudja rákényszeríteni termékét a

világra. Operációs rendszer szinten a technológiaváltáshoz 1-2 év szükséges. A jelenleg használatban lévő hardver többnyire nem is tudja fogadni az új operációs rendszert. Egy-egy váltáskor azt is figyelembe kell venni, hogy mi történt az elmúlt 10 évben, ezért a Windows 95 lassabban fog tért hódítani, mintha csak egy újabb verziójú MS-DOS lenne. Hosszú távon viszont aligha lehet elkerülni a bevezetését. Nem is a most megjelent változat rohamos elterjedésére számítok, hanem majd a kb. 2 év múlva elkészülőére.

Nem szabad messzemenő következtetéseket levonni a Windows 95 technológiai hibáiból sem. Ez visszatartó erő lehet egy darabig, de nem meghatározó. Tanulságos a Windows NT példája. Amikor megjelent az NT 3.1-es verziója, az iszonyúan gyenge dolog volt. A Microsoft cég azonban olyan erőket tudott felsorakoztatni, és annyi tőke állt mögötte, hogy idővel mégis lehetett az NT-ből valami működőt kihozni. Ráadásul a Unix-világban, az applikációs szerverek piacán, tehát nem is olyan területen, ahol ő volt az úr. Akkor hazai pályán miért ne tudna úgy focizni, hogy megtartsa vezető helyét?

● Ha a grafikus felhasználói felület ilyen mértékben domináns szerephez jut, egyáltalán mekkora játéktere marad a nem grafikus felhasználói felületre írt programoknak? A klasszikus DOS-os programok teljesen halálra vannak ítélve? Nem is érdemes velük foglalkozni?

— A hardvergyártók egyre erőteljesebb gépeket tudnak a grafikus felületek mögé állítani, ezért véleményem szerint az „ablakosodási" folyamatot nem lehet megállítani. Amíg lesznek applikációk, amelyek a grafikus felületre nincsenek felkészítve, addig persze lesz szerepe ilyen alternatívának, de egyre kevesebben vannak, akik az új szoftvereket „nem-Windows" környezetre fejlesztik. Ma a világon az installált PC-k 80%-a már rendelkezik grafikus felülettel, aki erre nem gondol, az a piac jelentős szegmenséből kizárja magát. A piac nincs tekintettel arra, hogy szíve szerint mit szeretne csinálni egy szoftverfejlesztő. Tulajdonképpen nem a technokraták határozzák meg a trendet, hanem a piaci igények. És a piac számára kulcskérdés a könnyű használat, ami egyenesen elvezet a grafikus alkalmazói felületekhez.

● Azt jelenti ez, hogy jelentősen átalakul és átértékelődik a programozói munka is? Más válhat benne fontossá?

— Sokan féltették a programozói munka színvonalát a grafikus programozói eszközök megjelenésétől. De mi történt valójában? Feltétlenül rosszabb programokat csinálnak azóta azok, akik grafikus eszközökkel dolgoznak? Nem. Tudni kell erről, hogy a grafikus eszközök rengeteg rutinszerűséget készen tálalnak, és az instant programozást teszik lehetővé, ami azt jelenti, hogy a programozó számára az applikáció-építés lesz a fő feladat, vagy az, hogy modulokat ír és azokat minél több helyre eljuttatja. Egy-egy problémán így kevesebb embernek kell dolgoznia, tehát hatékonyabb lesz a programkészítés, és ez is a számítástechnika elterjedését segíti.

Azt hiszem, hogy a hangsúly a programkód megírásáról a rendszertervek elkészítésére fog áthelyeződni, s végre eljutunk oda, hogy a feladatnak az érdemi részével mélyebben, szakszerűbben lehet foglalkozni.

● A Windows 95 megjelenése hogyan fogja befolyásolni a hálózati környezeteket?

— Kliens oldalon a Novell részéről megkapja a teljes támogatást, tehát a Windows 95 teljes értékű, 32 bites Netware kliense lesz a Novell-hálózatnak. Maga a Microsoft is felépít valami hálózati kommunikációt, amellyel bizonyos dolgokat kiválthat, kapcsolatokat építhet. Úgy látom azonban, hogy a Windows 95-ben nincs olyan szolgáltatás, amely a hálózati piacot jelen pillanatban komolyan megváltoztatná.

Ma még számítógép-hálózatokról beszélünk, de lehet, hogy holnap már csak általában a hálózatokról. És ez nem szójáték! Az a cél, hogy a tömegkommunikáció és a számítástechnika egyazon hálózaton éljen a telekommunikációval. A Novell jelen pillanatban azon dolgozik, hogy az évezredfordulóra mintegy 1 milliárd intelligens eszközt kapcsoljon hálózatba. Ha számítástechnikai szempontból nézzük, akkor addigra mintegy félmilliárd PC lesz a világon használatban. Ha csak azokat kötnénk be, akkor nem tudnánk elérni a célunkat, mert ott lesz a másik félmilliárd intelligens eszköz, amely processzorral rendelkezik, viszont grafikus felhasználói felülettel nem. A hálózatokhoz érve a Microsoftnak véget ér a birodalma.

● Tehát a Microsoftnak a hálózati fejlesztés irányába tett lépései nem jelentik azt, hogy meg is tudják ezt a piacot szerezni?

— A Microsoftnak ezen a területen nagy a lemaradása. Jelenleg kínált hálózati technológiái sokkal alacsonyabb színvonalúak, mint a forgalomban lévő alternatívák. A Microsoftnak a hálózati piacból való részesedése ma már meghaladja a Banyan piaci arányát, viszont két évvel mögötte jár a technológiában. A Microsoft piaci részesedése meg sem közelíti Novellét, annak csak 10%-a, technológiában pedig a Novell szerintem 3 évvel előbbre tart, mint a Microsoft. Ilyen helyzetben gyökeres változás nem várható a hálózatokon, s tegyük hozzá, hogy a Microsoft stratégiája a desktopra épül, nem pedig a hálózatra. Így semmi esélyét nem látom annak, hogy ezen a téren a Windows 95 áttörést hozhatna. A Windows NT már inkább játszhat bizonyos szerepet, de semmi esetre sem a jelenlegi állapotában.

● És milyen az NT viszonya a Windows 95-höz?

— Ha a felhasználó felteszi a kérdést, hogy mit vegyen inkább: Windows NT desktopot vagy Windows 95-öt, erre szerintem maga a Microsoft sem tud jól válaszolni. Ők maguk lassították az NT elterjedését azzal, hogy sok felhasználó nem vette meg a Windows NT-t, mert a Windows 95-re várt. Most pedig kiderül, hogy a kettő között olyan nagyon sok különbség nincs is.

● Talán rossz helyen teszem fel a kérdést, de kíváncsi vagyok a véleményére. Ha a Microsoft ilyen hatalmas erőket tud megmozgatni, és annyiféle irányba törekszik kiépíteni hadállásait, miért nem szán egy kicsivel több pénzt és energiát arra, hogy hibátlan programokat bocsásson ki, és kiküszöbölje az évek óta kritizált gyengeségeket (lásd lassúság) meg a kisebb-nagyobb fejlesztési problémákat?

— Magánvéleményem az, hogy amíg egy cég bevételeinek jelentősebb részét a piac megdolgozására, a marketingre fordítja, nem pedig a technológiára, addig nem is lehet jobbat elvárni tőle. Vannak technológiai irányultságú cégek (amilyen például a Novell is), és vannak más irányultságú cégek, amelyek jellemző képviselője a Microsoft. Bizonyos változás azért már a Microsoftnál is megfigyelhető, mert tendenciájában nézve egyre jobb szoftvereket hoznak ki, de még hosszú út áll előttük.

Faklen Pál

Programozni könnyen? Gyorsan?

A Windows áldott jó szíve...

Ez a cikk eredetileg könyvismertetésnek indult. Aki azonban elolvassa, egyet fog érteni velünk abban, hogy témája és megközelítési módja miatt előrehoztuk ide, A hónap témájának általánosabb jellegű összeállításába. A Könyvespolc rovatban (59. o.) szintén Windows-felületen futtatandó alkalmazói programok készítésével foglalkozó könyveket mutatunk be. De az ismerkedést érdemes ezzel az írással kezdeni...

Nagyon elszánt programozónak kell lennie annak, aki MINDAZT meg akarja jegyezni, ami ebben a könyvben* található. De talán nem is ezzel a céllal írták. A lexikon sem arra való, hogy egész tartalmát megtanuljuk, hanem hogy utánanézhessünk mindannak, amire éppen szükségünk van.

Maguk a szerzők sem az OWL-ben (Object Windows Library) való programozás konkurens módszereként ajánlják ezt a programozási stílust. Szerintük azért érdemes előbb a „sima" C nyelvű programíráson keresztül megismerkedni a windowsos alkalmazásokkal, hogy jobban megértsük azok működését. És hogy lássuk, mi mindentől menthet meg bennünket az objektumorientált programozás... — tehetnénk hozzá némi iróniával. Egyébként nem a Borland az egyetlen, amely OOP-re alapozott rendszerekkel igyekszik megkönnyíteni a programírást. A Microsoft hasonló céllal adta közre Microsoft Foundation Class (MFC) elnevezésű objektumkönyvtárát, amely természetesen éppúgy jótékony lepellel fedi el a Windows működését, mint a Borland rendszerei.

Az elvarázsolt világ valósága

Mint tudjuk, maga a Windows is objektumorientált elvek alapján működik, ezzel képes biztosítani számos újszerű lehetőséget saját maga és a felhasználók számára. Az OOP-n keresztül elérhető, hogy az egyes alkotóelemek viszonylagos függetlenséget élvezzenek. Előre definiálhatók bizonyos képernyőelemek (gombok, különböző „okos" ablakok), működésük jellegzetességeivel együtt. Környezetfüggetlenné tehetők a programok, mert a Windows a perifériák alacsonyszintű kezelését egységes módon el tudja intézni.



Ez volna hát az, ami miatt a Windows alatti programozás alapvetően másféle gondolkodást igényel? Talán inkább magának a szemléletmódnak a megváltozása. Próbáljunk utánagondolni, vajon miért.

A DOS alatt sokkal világosabban elkülönültek a szerepek. Amikor egy program futott, az kisajátította magának az egész számítógépet, a futás idejére övé volt az egész világ — de saját magának kellett gondoskodnia jóformán mindenről. Az operációs rendszer csak olyankor jutott szóhoz, amikor a program már befejezte a dolgát. Vagy amikor kikölcsönöztek tőle egy-egy rutint, mert a programnak szüksége volt a DOS vagy a BIOS valamelyik szolgáltatására.

A Windows sokkal nagyobb mértékben hajlandó besegíteni a programok futásába. Kölcsönadja nekik saját erőforrásait, könyvtárait, függvényeit, mindenét... Segít a megjelenítésben, a felhasználóval való kapcsolattartásban, üzenetek értelmezésében és továbbításában. Tulajdonképpen elválasztani is nehéz, hogy hol végződik az alkalmazói program saját tevékenysége, és hol kezdődik a Windows közreműködése. A Windows áldott jó szíve azonban nem egyetlen programé, hanem mindegyiké. Mindegyik megkaphatja tőle azt, amit csak akar — ha kivárja a sorát.

A Windows a nagy illúziók rendszere. (Már abban is, hogy operációs rendszernek mutatja magát, pedig nem az.) Windows alatt egyszerre több program hiheti magáról, hogy éppen őt futtatják. Ha azonban túlságosan beleélné magát, hogy ő a dudás a csárdában, alaposan pórul járhat: elvehetik tőle még a levegőt is.

A multiprogramozás látszata

A Windows látszólag „multiztatja" a programokat, pedig voltaképpen mindig csak egyikük fut egyszerre. Az illúziókeltés szempontjából nem is ez a lényeg, hanem a Windowsnak az a képessége, hogy a futtatást darabokra tudja szabdalni: szinte bármikor félbeszakíthatja és újraindíthatja őket. Ez pedig a szkeptikusabb felhasználókban is táplálhatja a multiprogramozás illúzióját. Ha egy program például a földgömb forgását mímeli a képernyőn, akkor ugyanannak a programnak különböző példányai különböző fázisokban mutathatják Földünk forgását. A mozi és a tévé varázslata után újabb bűvészmutatványnak lehetünk tanúi: egyetlen képernyőn akár több példányban is megelevenedhet a látszat valósága.

A valóságos multiprogramozás tulajdonképpen már nem is olyan fontos. Ha valamilyen külső vagy belső esemény hatására bármelyik program futása

*Benkő Tiborné—Kiss Zoltán—Kuzmina Jekatyerina—dr. Tamás Péter—Tóth Bertalan: Könnyű a Windows-t programozni!? ComputerBooks, 1995 (2. kiadás), 628 oldal, 1683 Ft (lemezmellékléttel).

megszakadhat vagy folytatódhat, akkor már az is megoldható, hogy a tetszhalálban egymás mellett élő programok kommunikáljanak egymással, vagy éppen az egyik program által érzékelt esemény vezérelje a másik futásának folytatását.

Nem nehéz belátni, hogy csírájában ebben a koncepcióban már benne rejlik az ember-gép kapcsolatnak és a programok közötti kooperációnak minőségileg magasabb szintje.

Ablak ablak hátán

A külső szemlélő számára a programoknak ez a sokat ígérő együttműködése úgy realizálódik, hogy minden, látszólag „futó" programnak van egy ablaka (extrém esetben ez az ablak egy apró ikonná korcsosulhat), és hol az egyik program változtatható aktívvá, hol a másik. Ha valamilyen információra van szüksége a programnak, akkor szól - azért kapta meg a képernyőnek egy téglalapnyi területét. Ezen a sajátjaként birtokolt hirdetőtáblán közli a felhasználóval óhajait, itt kérhet további eligazítást: néhány gombot dug a felhasználó orra alá, hogy válasszon, vagy egy kisebbfajta kérdőív kitöltésére noszogatja.

Az új szisztémában ugrásszerűen megnőtt az interaktivitást biztosító eszközöknek, a képernyőnek, az egérnek és a billentyűzetnek a jelentősége. Megemlíthetnénk még a fényceruzát és a hangérzékelőket is, annak ellenére, hogy ezek jelenleg (ki tudja, már vagy még) kevés helyen játszanak lényeges szerepet. Egy kis túlzással akár azt is mondhatjuk, hogy a lineáris programfolyamokat felváltotta valami olyan programozási szituációnak a leírása, ahol — potenciálisan legalábbis — minden mindennel összefügg. Az egyes szituációkon belül is, de a szituációk között is fennáll a kölcsönhatás: átmehet az egyik a másikba, akadályozhatja vagy elősegítheti annak kibontakozását.

Velem mindig történik valami...

Szinte azt mondhatnánk, hogy a determinisztikusan vezérelt folyamatokat felváltotta a programozásban a véletlenszerűen bekövetkező események uralma. Azt sem szabad elfelejtenünk, hogy látszólag apró kis történések is tevékenységek egész sorozatát vonhatják maguk után. Az egérkurzor odébbhúzása például gyakran csak véletlenszerűen következik be, nyomában mégis gyors egymásutánban újra és újra át kell festeni a képernyőt. Erre kényszerít az események követésének kérlelhetetlen logikája, mivel a koordináták folytonosan változó állapotát is híven tükrözni kell — ha nem is folyamatosan, de rendszeres időközönként.

Az eseményekről különböző üzenetek értesítik a programokat. Állandóan záporoznak az üzenetek, és ezek közül minden program maga választja ki, hogy melyik szólhat neki.

Szegény ártatlan program valóban nehéz környezetbe került. Bármely pillanatban leüthetnek egy billentyűt, odébb tolhatnak egy egeret, pöttyenthetnek egyik vagy másik egérgombon, vagy éppen oldalba billentheti az operációs rendszer valamelyik rutinja. Neki pedig számításba kellene venni minden előforduló helyzetben minden lehetséges és lehetetlen eseményt, hogy megfelelően reagálni tudjon a neki szóló üzenetekre, a többit pedig elengedje a füle mellett.

Valóban jól jön ilyenkor a Windows önzetlen segítsége. Amelyik üzenettel nem tud mit kezdeni, azt például továbbküldheti a Windows erre a célra rendszeresített „postabontójának", üzenet formájában várva az esetleges választ...

Konklúzió?

Nos, könnyű vagy nehéz olyan programokat írni, amelyek ilyen szituációban helyt tudnak állni?

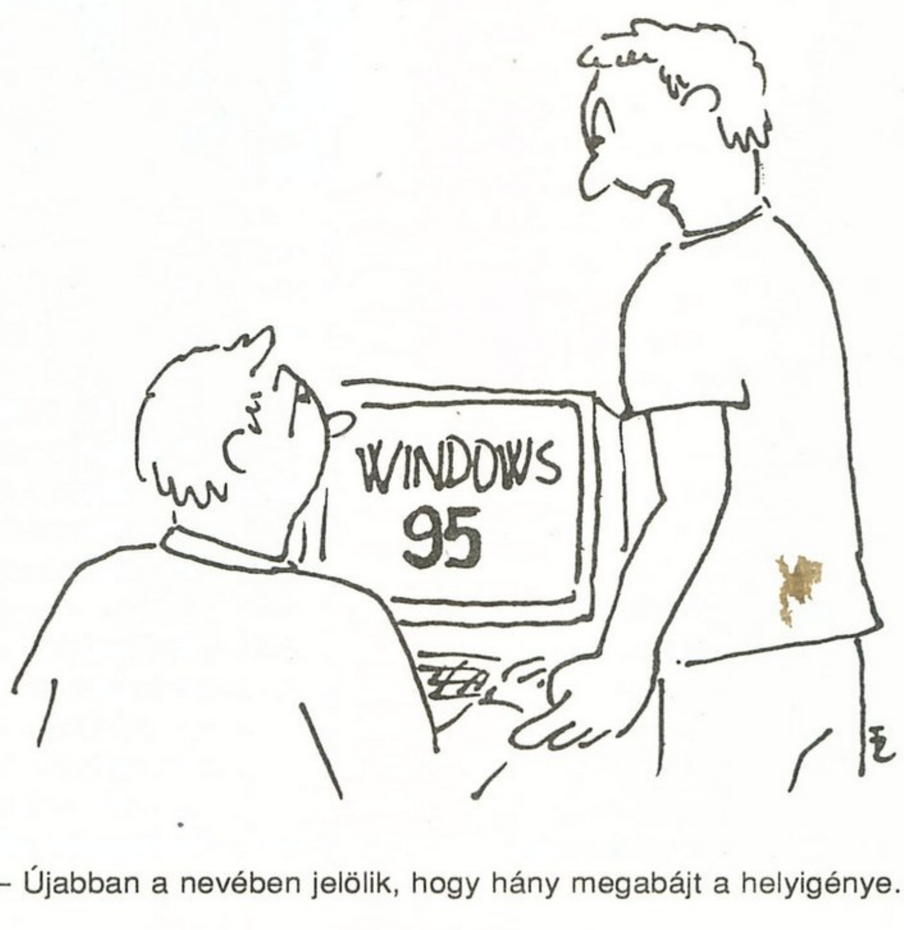
Hogy nem lehetetlen, azt bizonyítja a könyv lemezmellékletének majd félszáz programja, a legkülönbözőbb helyzetekben mutatva meg, hogy miként. (Ha valakinek nem áll rendelkezésére a megfelelő fordítóprogram, jó, ha tudja: a Windows-zal futtatható EXE program is megtalálható mellettük.)

Még a segítség is rendelkezésünkre áll a programíráshoz bármely pillanatban, de tudnunk kell, mikor hogyan kérhetjük. Csupán néhány száz konstans üzenet értelmét és használati módját kell elsajátítanunk, meg a Windows-alkalmazások programozói felületének (Application Programming Interface, API) rendszerfüggvényeit kiismernünk, hogy neki tudjunk fogni...

A könyv folyamatos végigolvasása csak mazochistáknak ajánlható, ők azonban élvezni fogják. Viszont már az első három fejezet elolvasása mindenkit feljogosít arra, hogy elkezdje böngészni a mellékletben közölt programok forráslistáit, s rá fog jönni, hogy ez sem boszorkányság. Kellő elszántsággal viszonylag hamar eljut oda, hogy tájékozódni tud a programban, és látja, mi miért történik. Tematikusan azután később is kiválasztható a könyvből egy-egy fejezet, olvasni szabad, de az olvasottakat szigorúan tilos megjegyezni. Harmadik olvasatra már teljes magabiztossággal lapozhatunk a könyvben előre-hátra, s olyan benyomásunk támad, mintha kezdenénk kapiskálni...

A könyv hasznossága vitathatatlan, s még azt is meg merném kockáztatni, hogy a tálalt anyag túlnyomó része nélkülözhetetlen a windowsos alkalmazói programok megértéséhez. Azt azonban kötve hiszem, hogy sokan lennének, akik az olvasottak alapján elhatároznák, hogy ők ezután csak ilyen módon írnak programokat...

Vargha Dénes



— Újabban a nevében jelölik, hogy hány megabájt a helyigénye.

[csinnadratta, trallalla] „Hölgyeim és uraim!” [csinnadratta, trallalla]

„Következik: Windows 95...” [taps]

Meglehetősen sokat, több mint egy évet késett a Microsoft új Windows-verziója. Még Windows 4.0-ként fogant, embrionális fejlődése idején keresztelték el Chicagónak, végül Windows 95 néven született meg. Az egymást követő 8 (!) béta-verzió után 1995 májusa óta kínálják a viszonteladóknak és a legnagyobb ügyfeleknek a Windows 95 Preview csomagot, amely a közben még kijavítandó hibáktól eltekintve feltehetően azonos az augusztus végén forgalomba kerülő végleges kereskedelmi változattal. A magyar kiadás ezek után az év utolsó hónapjaiban készülhet el. A Microsoft Magyarország Kft. jóvoltából mi is kipróbálhattuk az új Windowst, hogy tapasztalatainkról beszámoljunk olvasóinknak.

Előzetesen a Microsoft a következőkben foglalta össze a Windows 3.1-est felváltani (sőt jóval többre) hivatott új rendszerrel szemben támasztandó követelményeket:

— Kompatibilitás a DOS/Windows 3.0, 3.1 alatt futó alkalmazásokkal.

— Egyszerű, problémamentes installálás.

— Teljesen 32 bites rendszer, valódi (preemptív) multitaszking.

— A karakter alapú DOS parancsfelület kiküszöbölése.

— Hardverelemek egyszerű installálása „plug & play” elven.

— Beépített hálózati kapcsolat.

— Hosszú fájlnevek használata.

— Új, hatékonyabb grafikus felület.

— Egyszerűbb rendszerkarbantartás és adminisztráció.

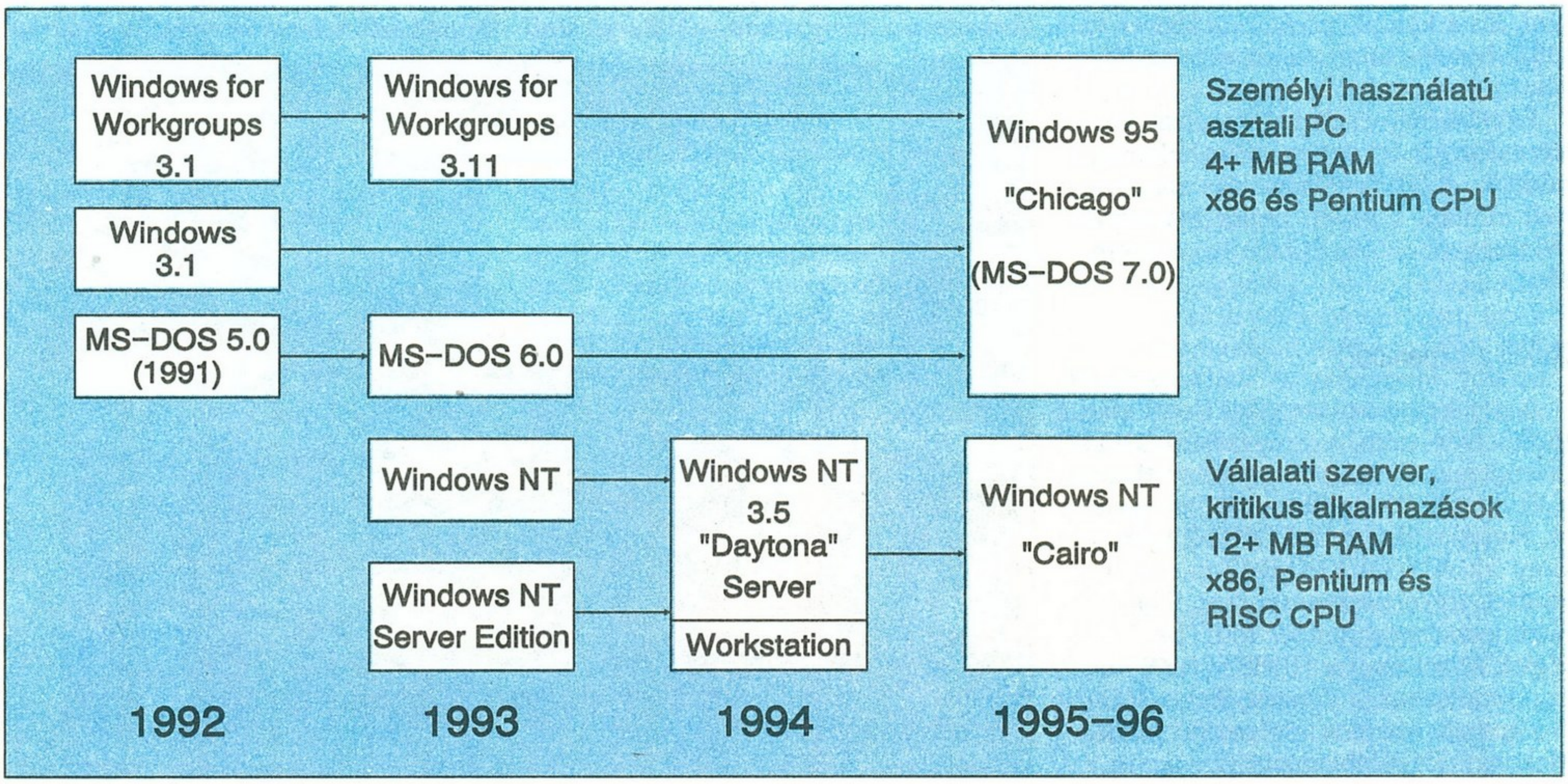
Nézzük meg, mi valósult meg az ígéretekből.

Installálás

A szoftvert 3,5"-es floppylemezekről vagy CD-ről installálhatjuk. A floppy nem a megszokott 1,44 MB formátumú, hanem 1,6 MB-ra formázott. Így valamivel kevesebb lemez kell hozzá, ezáltal csökkennek a Microsoft költségei, és mellesleg az eddigi programokkal nem másolható. (Persze csak addig, amíg nem írnak vagy nem szereznek be hozzá ilyen másolóprogramot.)

A CD-ről egyszerűbb a telepítés, több nemzeti verzió közül választhatunk. A Preview csomagban nem, de az időközben hozzánk eljutott kelet-európai béta-verzióban magyar változat is van, bár annak magyarsága egyelőre még olyan, mint 30 éve külföldre szakadt egynémely hazánkfiáé.

A Windows 95 upgrade jellegű program, vagyis üres merevlemezre nem telepíthető. Lennie kell rajta legalább MS-DOS 3.3-nak (32 MB-nál nagyobb partícióval), vagy Windows 3.x-nek. A hardverszükséglet minimálisan 386DX processzor és 4 MB RAM. A működéshez emellett legalább 14 MB szabad memóriaterület kell, így 4 MB RAM-mal a merevlemezen az átmeneti állomány (swap fájl) nagyobb, mint 10 MB.



A fájlok helyigénye attól is függ, hogy volt-e előtte a gépen Windows, és hogy mit teszünk fel az újból. A képernyőadapter VGA-nál kezdődik, az EGA-t már nem támogatja.

Amennyiben van a gépen 3.x Windows, mindenképpen érdemes onnan indítani a Setupot. El kell döntenünk, hogy felégetve magunk mögött a hidakat, a régi MS Windowst rögtön lecseréljük-e a 95-ösre, vagy óvatos duhajként az előzőt megtartva telepítjük. Az első esetben automatikusan átveszi a már beillesztett alkalmazásokat, a második esetben viszont mindent újra kell telepítenünk benne, és lényegesen több, minimálisan 40 MB szabad hely kell a merevlemezen. A kész Windows 95 nem foglal el ennyit, csak kb. 30-at, de a Telepítő Varázslónak kell a plusz hely. (Hát igen, a Windows körül egyre szaporodnak a varázslók, tippadók, koboldok és egyéb vajákosok.)

A Telepítő automatikusan a meglévő Windows-könyvtárat célozza meg. Ha azt meg akarjuk régi formájában őrizni, írjunk be új könyvtárnevet. A telepítő következő kérdése, hogy mennyit tegyen fel az új Windows-változatból:

— Typical (Szokásos) — amit a rendelkezésre álló lemezterület függvényében fontosnak tart.

— Portable (Hordozható gépekre).

— Compact (Tömör) — amit inkább minimumnak lehet értelmezni.

— Custom (Egyéni) — válogass a saját szakálladra.

A Setuppal készíthetünk egy indító floppylemezt is, amellyel elindíthatjuk a rendszert, ha azt a merevlemezről valamilyen ok miatt nem tudjuk betölteni. Ez a különböző Unixokban és az OS/2-ben is ismert, hasznos szolgáltatás.

Ezután már csak időzónánkat és a használni kívánt nyomtatókat kell megadnunk. A fájlok felmásolása után újra kell indítani a gépet. Ekkor már az új Windows indul el, és befejezi a telepítést.

Mi lett a gépemből?

A gép bekapcsolása vagy Reset után többé már nem a megszokott DOS prompt jelenik meg, hanem közvetlenül a Windows 95 grafikus felülete indul el. Nincs szüksége a CONFIG.SYS-re vagy az AUTOEXEC.BAT-ra, bár ha vannak, figyelembe veszi őket. Saját 32 bites eszközmeghajtóit tölti be, s a régiek közül csak a HIMEM.SYS kell az induláshoz. Amennyiben DOS promptban akarunk dolgozni, kérhetünk egy DOS ablakot, vagy pedig a Shutdown (Kikapcsolás) parancsban a rendszerleállítás, újraindítás és MS-DOS módba való kilépés közül az utóbbit kell választanunk.

A Windows 95 új MS-DOS-verziót is telepít, amely a VER parancsra „Windows 95 [ver. 4.00]"-ként mutatkozik be, de az IO.SYS-ben megtalálható az „MS-DOS Version 7" copyright jelzés is. Az új DOS nem teljes, olyan értelemben, hogy a korábban megszokott DOS-parancsok nem mindegyike kapott helyet az új verzióban.

Amennyiben a telepítéskor új könyvtárnevet adtunk meg, nemcsak az előző MS Windowst, hanem a DOS-t is megőrzi. A rendszerfájloknak, a CONFIG.SYS-nek és az AUTOEXEC.BAT-nak DOS kiterjesztést ad, ezután helyezi el a főkönyvtárban az új IO.SYS-t, MSDOS.SYS-t és COMMAND.COM-ot. A CONFIG.SYS-ben és az AUTOEXEC.BAT-ban törli a feleslegessé vált parancsokat (ilyen például a FASTOPEN vagy a BUFFERS), amelyekben pedig nem biztos, azokat korrekt módon, a „REM – by Windows 95 – " megjegyzéssel iktatja ki. Így áll elő az új CONFIG.SYS és AUTOEXEC.BAT. Csak akkor van rájuk szükség, ha a DOS módban (nem DOS ablakban!) is akarunk dolgozni. Akkor viszont mindenképpen, mert a Win95 meghajtói csak saját környezetükben dolgoznak. Ha kilépünk a DOS-ba, „csupasz" gépünk van, hacsak nem töltöttük be a meghajtókat induláskor.

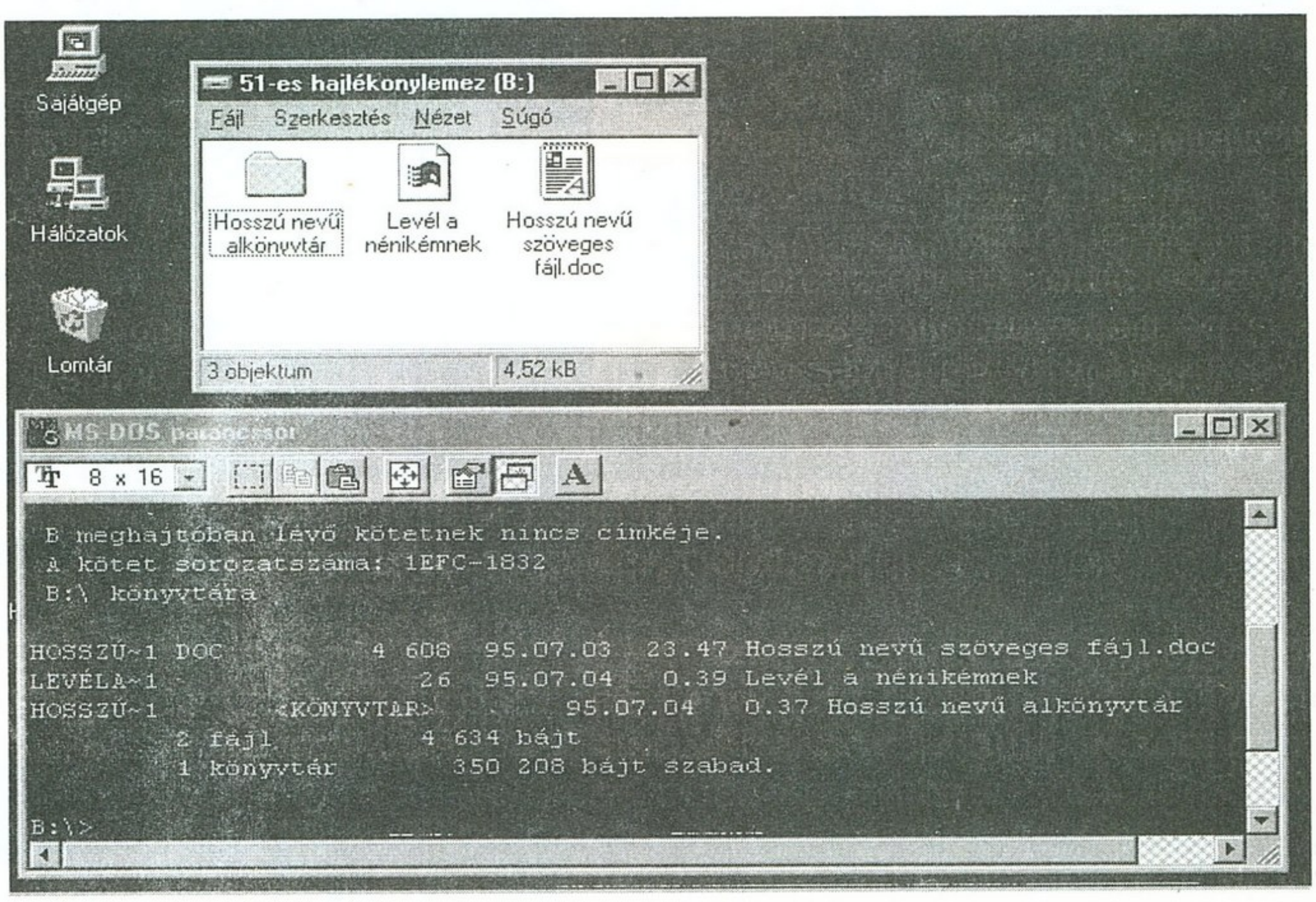
A magyar Win95-tel járó MS-DOS is magyar. Annak, aki idestova 10 éve használja a DOS mindenféle változatát, egyelőre még furcsa érzés, ha a könyvtárlistában a megszokott <DIR> jelzés helyett az szerepel, hogy <KÖNYVTÁR>.

Bekapcsolás után, a Starting Windows 95... (A Windows 95 indítása...) felirat megjelenésekor az F4 gombot lenyomva az előző MS-DOS verziót tölti be (ha telepítéskor megtartottuk), visszanevezi a rendszerfájlokat, és a programok a Win95 előtti környezetben futhatnak, használhatjuk a korábbi Windowst is. A Win95-öshöz mindenképpen kell az új DOS: ha más verzió alól akarjuk indítani, udvariasan közli, hogy az MS-DOS 6.x és egyéb korábbi verziók alatt nem működik.

A betöltés problémáit segít felderíteni az F8 gomb, amelyet szintén az induló felirat megjelenésekor nyomhatunk meg. A megjelenő menüben többféle indítás közül lehet választani. A normál betöltésen kívül kérhetjük az indulás naplózását, ekkor egy BOOTLOG.TXT fájlba írja, hogy mely részeket sikerült rendben indítania.

Választhatjuk a Safe mode-ot (Biztonságos módot), amikor csak a legszükségesebb meghajtókat tölti be, valamint azt, hogy megerősítést kérjen minden egyes fájl betöltéséhez, és hogy ne töltse be a grafikus interfészt, csak a DOS-t.

Az installálás lassú gépen akár egy óráig is eltart, de két kisebb hibát leszámítva komolyabb problémába nem ütköztem. Az első hiba akkor lépett fel, amikor a Novell DOS 7 mellé telepítettem. Az F4 gombra közölte, hogy az előző verzió nem kompatibilis, és elindította az újat, pedig a szintén béta-verziójú Windows 95 Resource Kit szerint mennie kellett volna. A fájlok kézi visszanevezgetésével persze



el lehetett indítani, de ez nem valami elegáns. Ha csak a parancsprompt betöltését kértem, a régi Windows természetesen használható volt. A régi és az új együttélése nyilván a biztonságos és kényelmes megismerést szolgálja. A különcök (oktatók, szakírók stb.) kivételével a felhasználók többsége valószínűleg hamar eldönti, hogy átáll-e az új Windowsra, vagy megmarad a réginél. Nekik két DOS-t és két Windowst a gépen tartani valóban helypazarlás lenne.

A második hiba ahhoz kapcsolódott, hogy a program a grafikus interfészt — akárcsak az előző verziónál — egyszerűen a WIN.COM-mal indítja. Miután a CD-ről felmásolta a fájlokat, és felszólított, hogy indítsam újra a gépet, a Reset után szépen az előző 3.1-es Windows jelentkezett be. A telepítő betette ugyan a PATH változóba (az első helyre) a Win95 könyvtárát, de nekem a főkönyvtárban volt egy WIN.BAT-om, még az előző verzióhoz. Mivel a keresésben az aktuális könyvtár megelőzi a PATH változó tartalmát, a régi Windowst indította. Egyszerű eset, a WIN.BAT átnevezésével megoldható volt, de elképzelhető, hogy ettől egy kezdő felhasználó zavarba jön.

Kompatibilitás

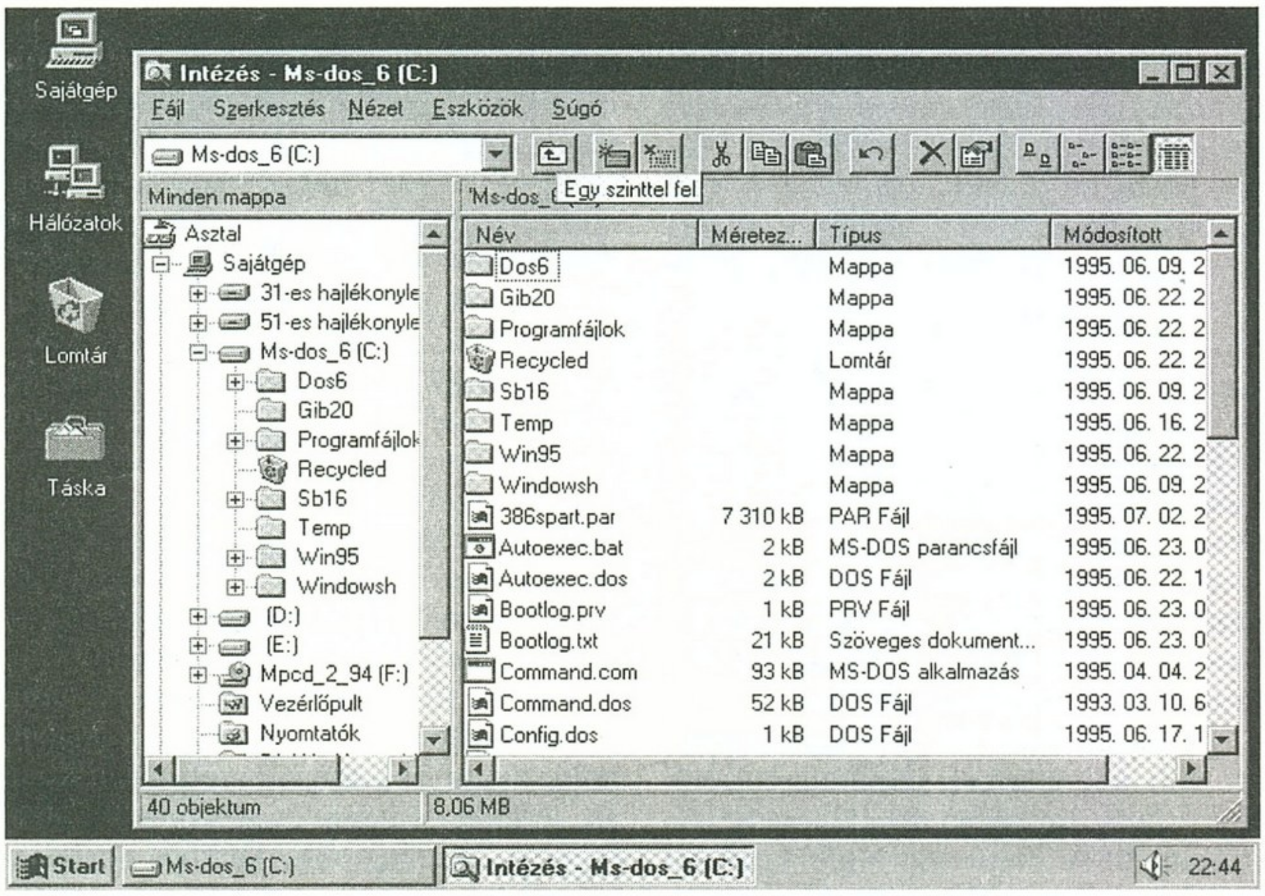
Erre a jelek szerint ügyeltek a fejlesztők, mert valamennyi általam kipróbált MS Windows 3.1-hez készült alkalmazás gond nélkül élt rajta. A Microsoft szerint pedig jóval kisebb az új Windows rendszerből nem futtatható DOS programok száma, mint amennyi a korábbi Windows-verzióknál volt.

Hány biten?

A „teljesen" 32 bites rendszer természetesen tartalmaz 16 bites kódú, főként Assemblyben írt részeket. A 32 bites kód nagyobb méretű, általában gyorsabb is, de ez már a körülményektől és a fordítótól függ. A legfontosabb részek, a kernel, a GDI és a user modul 32 bites, főként a korábbi 16 bites Windows-alkalmazásokat kiszolgáló részben maradtak 16 bites modulok.

„Valódi" multitaszk

Én is próbáltam a multitaszkra jó magyar szót találni, de egyik sem az igazi. A multiprogramozás nem jó, mert a felhasználó itt nem programoz semmit. Többfeladatúság (egyik laptársunk kedvence), szó szerinti fordításban igaz, mert a task feladatot is jelent, csak



éppen az értelme vész el a számítógépes környezetben. Hétköznapi analógiával talán a „szimultán" szó fejezné ki legjobban a multitask fogalmát (lásd a sakkszimultánt). Jobb híján egyelőre tehát multitaszknak hívom az operációs rendszernek azt a tulajdonságát, hogy egyidejűleg — esetünkben időosztásos elven — több felhasználói program futtatását is lehetővé teszi.

Eddig a Windows hiányosságaként említették, hogy nem „valódi" multitaszkkal dolgozik. A valódi multitaszk ebben az értelmezésben ugyanis azt jelenti, hogy amikor a neki kiszabott CPU-idő letelt, az operációs rendszer elveszi a processzort a felhasználói programtól, és egy másiknak adja oda. Ez a preemptív (preemptive), szó szerint elővételi multitaszk, mert az operációs rendszernek előjoga van a CPU-ra.

A Windows 3.1 és a többi korábbi verzió e helyett az együttműködő, kooperatív (cooperative) módszert használja. Ebben az operációs rendszer üzenetet küld a futó programnak, hogy ha letelt az ideje, a program adja vissza a vezérlést az operációs rendszernek. Meg is teszi, ha tudja. Előfordulhat viszont, hogy éppen olyan műveletet végez, amelyet nem szakíthat félbe, vagy hiba történt, és a program elszállt. Ha pedig az operációs rendszer nem kapja vissza a vezérlést, akkor csak a Reset vagy a Ctrl-Alt-Del segít.

A Win95 az új, 32 bites (Win32) alkalmazásokat már preemptív módon futtatja, az eddigi 16 biteseket (Win16) azonban a kompatibilitás megőrzése miatt továbbra is kooperatív módon. Vagyis az eddigi, 16 bites alkalmazások esetében nem számíthatunk stabilabb működésre. A preemptív módszer a nagyobb megbízhatóságon kívül gyorsabb, gördülékenyebb felhasználói felületet eredményez, amit egyszerűen ki is próbálhatunk. A Windows 3.1 Filekezelőjében indítsuk el egy floppy formázását (azért éppen ezt, mert viszonylag hosszú ideig tart). Közvetlenül ezután lépjünk be a Paintbrushba, és folyamatosan lenyomva tartott egérgombbal rajzoljunk. A formázás ezért leáll, a floppy lámpája kialszik. A Paintbrush, mivel folyamatos inputot kap, nem ereszti el a processzort, így a File-kezelő nem tud ráállni a következő sáv formázására. Amint felengedjük az egér gombját, a Paintbrush is elereszti a CPU-t, a floppy lámpája kigyullad, és folytatódik a formázás.

Ugyanezt eljátszva a Win95-ben — ott Paintnek hívják a beépített rajzolót — a formázás nem áll meg, bár rajzoláskor a vonal kissé ugrálva követi a ceruzát, de ennek a mértéke már a CPU sebességétől függ.

A DOS-alkalmazásoknál alapfeltevés, hogy egyedül van a gépen, tőle csak elvenni lehet a processzort, így a DOS-alkalmazások már a korábbi MS Windowsokban is preemptív módon futnak. Ezt a Windows a megszakítások átvételével oldja meg. Amint egy DOS-alkalmazás megszakítást kér, vagy hozzá megszakítás érkezik, a Windows el tudja venni a CPU-t.

Profi(bb) kinézet

A felhasználók számára a Win95 legnagyobb újdonsága természetesen a grafikus felület, ami alig emlékeztet a 3.x verziók felületére. Megváltozott a

képernyő minden alapeleme és a grafikus felület szerkezete. Ami a szerkezeti változásokat illeti, nincs többé sem Programkezelő, sem File-kezelő. Pontosabban, ha a telepítéskor az egyéni válogatásba lépünk, akkor van egy olyan menüpont, amelyben a Win3.1 és a Win95 megjelenítés között választhatunk. A Win3.1 esetén lesz egy Programkezelő és egy File-kezelő ablakunk, de mert nem valami jól illeszkednek az új környezetbe, tényleg csak a nagyon óvatosaknak érdemes ezt választaniuk. (Vagy talán még nekik sem.)

Az installálás után a 4. oldalon látható kép fogad minket az új Windowsban. A pontos kinézet persze attól is függ, hogy mit pakoltunk fel.

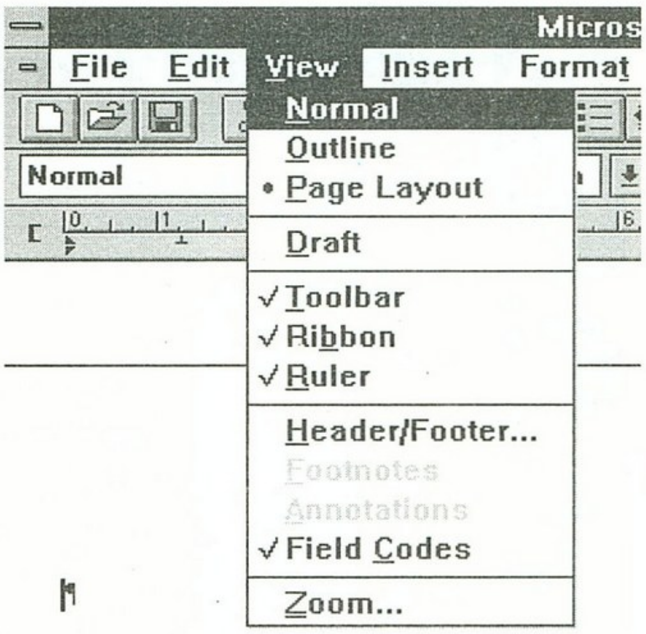
Az asztalon (desktop) mindenképpen ott van a gépünket szimbolizáló My Computer (Sajátgép) és a Recycle Bin (Lomtár). A Network Neighborhood (Hálózatok) csak akkor, ha gépünk hálózati kártyát is tartalmaz. A képernyő alján a Taskbar (Tálca) található.

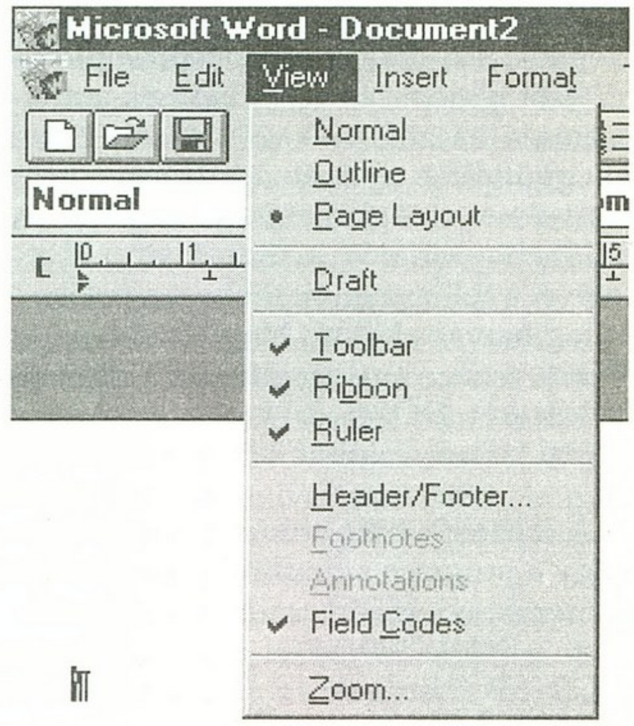
A Sajátgép ikonon kétszer kattintva megjelenik a Sajátgép ablak, amelyben egy-egy ikonnal szerepel minden háttértároló. Ha meg akarjuk nézni a tartalmát, csak rá kell kattintani, és megnyitja a hozzá tartozó ablakot.

Installáláskor minden merevlemezpartíción létrehoz egy Recycled könyvtárat, ennek a kezelését végzi az asztalon a Recycle Bin (Lomtár) ikonnal szereplő program. Amikor a Win95-ön belül törlünk egy fájlt vagy könyvtárt, az automatikusan a Recycled könyvtárba kerül át. Előnye: minden megvan addig, amíg a Lomtár programba lépve ki nem ürítjük, azaz véglegesen nem töröljük. Hátránya: amíg ezt nem tesszük, a merevlemezen nem szabadul fel a hely. A DOS ablakban (vagyis a parancssorban, mert ez a hivatalos neve, egyébként úgy néz ki, mint egy ablak), szóval a DOS-ban kiadott DEL parancsra ezt a raktározást nem csinálja, és a floppylemezeken végrehajtott törléseknél sem.

A képernyő alján lévő Taskbar (Tálca) a Win95 legfontosabb kezelőszerve. Induláskor csak a Start gomb és a jobb sarokban az óra van rajta. A Start gombot megnyomva jelenik meg a főmenü, annak pontjaiból újabb menük nyílnak. Itt már nem kell kattintgatni, elég csak rátolni az egeret a megfelelő menüpontra, és a hozzá tartozó almenü (ha van) automatikusan feltűnik. A menük tartalma természetesen változtatható, minden gyakran használt programot érdemes ide felvenni.

A legalsó menüpont a Kikapcsolás (Shutdown). A gép kikapcsolása vagy





Reset előtt ki kell adni ezt a parancsot. A Tálcán minden megnyitott ablakot egy gomb jelez. Ha be van nyomva, akkor a hozzá tartozó ablak az aktív. Az előző Windowsban az összecsukott ablak egy ikont hagy hátra, amely dokumentumablaknál az alkalmazás ablakának aljára, programablak esetén pedig az asztallapra kerül. Mindkét helyen lefedhetik azokat más, nyitott ablakok, ezért az újbóli kinyitáshoz néha keresőakciót kell indítani. (Persze a Ctrl+Esc-pel meg lehet hívni a Futó Programokat, ahol át lehet váltani, de ez a lehetőség a képernyőn nem látszik.) A Win95-ben az összecsukott ablakokat a Tálcán lévő gombjukkal lehet újra kinyitni, az ábrán a kinyitott, de nem aktív Sajátgép ablak gombja mellett a becsukott Számológép gomb van a Tálcán.

Ahogy az a 4. oldali képernyőábrán is látszik, megváltozott az ablakok és a kezelőikonok megjelenése is. Az előző verziónál a bal felső sarokban lévő, mínuszjelre emlékeztető Control ikon helyét átvette az alkalmazás ikonja, de a vele hívható Control menü ugyanaz maradt. A jobb oldalra három új ikon került. Balról az első csukja össze az ablakot, a vonal a Tálcát szimbolizálja, mivel a becsukott ablakok ott érhetők el. A középső ikon az ablakot maximális méretűre nagyítja, végül a jobb szélső bezárja az ablakot, kilép a hozzá tartozó alkalmazásból. Ez felel meg az előző verziónál a dupla kattintásnak a Control ikonon. Jó, hogy ezt a funkciót külön ikonra tették, de az elhelyezése nem szerencsés. A jobb szélen a maximálisra nagyító ikon volt, és az új elrendezésben kinagyítás helyett — a megszokás miatt — nemegyszer bezártam a futó alkalmazást.

A képernyőelemek és a rendszerfontok változása minden Windows-alkalmazásra hatással van. Az ezen az oldalon lévő ábrán példaként a Winword 2.0 egyik menüje látszik a Win3.1 és a Win95 alatt futtatva.

Új tulajdonságok

A Win95 kezelésének egyik újdonsága, hogy a jobb oldali egérgombbal szinte minden képernyőobjektumnál hívható egy menü, amely a neki megfelelő parancsokon kívül mindig tartalmaz egy Tulajdonságok (Properties) menüpontot. Ebben a rá vonatkozó paraméterek megváltoztathatók. Így például az asztallapon kattintva a jobb egérgombbal a Tulajdonságokban beállítható a tapéta, a képernyőkímélő, a szín, a felbontás stb. Mindez természetesen a régi módon a Start menü Beállítások/Vezérlőpult/Képernyő paranccsal is megtehető, de az első módszer sokkal kényelmesebb és gyorsabb. Apropó, felbontás. Az új, beépített virtuális eszközmeghajtóknak köszönhetően azonnal kipróbálható mondjuk egy másik képfelbontás, nem csak újraindítás után, mint korábban.

A Programkezelő és a File-kezelő hiánya a régi Windows-felhasználókat kezdetben mindenképpen zavarni fogja.

A File-kezelő funkcióit azonos meghajtón belül maradva a Sajátgép ablak vette át, itt lehet egy programfájlt indítani, új könyvtárat létrehozni, floppyt formázni stb. Több meghajtóhoz egyszerre hozzáférni a Start menü/Programok pontban lévő Windows Explorerrel (Windows Intéző) lehet. (Az Explorert én inkább Keresőnek fordítanám.) A 15. oldali képernyőábra az Intéző ablakát mutatja. A könyvtárra és fájlpanelre osztott forma emlékeztet a File-kezelőre, de ezt a megjelenést itt nem lehet megváltoztatni. A eszközsor min-

den ablakban egységes formában bekapcsolható, legfeljebb nincs mindig minden ikonnak funkciója. A némi várakozás után megjelenő ikoncímkék az alkalmazások (mint a WinWord 6.0 vagy Excel 5.0) után itt is megjelentek. A gördítősávok csúszkáinak hossza arányos a látható/nem látható területtel, amiből a nem látszó rész nagyságára lehet következtetni. Ez csak a Windowsban új, a GEM grafikus felületen már 10 éve kitalálták. Az Intézőben lehet a szokásos, meghajtók közötti fájlműveleteket elvégezni.

Érdekesség: teljes floppylemez másolásához sem a Sajátgép ablakban, sem az Intézőben nincs parancs vagy ikon. A Help azt mondja: lépjünk be egy DOS ablakba, és diskcopy x: y: ...

A Programkezelő feladatait részben az asztal (desktop), részben a Tálca vette át. A Tálca jobb oldali egérgombbal hívott menüjében lehet a megnyitott ablakokat különböző formában elrendezni.

Sokat javítottak a Helpen (Súgó). A korábbi Windowsban az F1-re megjelenő Súgó képernyő rendre eltakarta a munkaablakot, most ha csak lehetséges, mellette tűnik fel. Tömörebb lett a szöveg, és nőtt a tárgymutatóba felvett kifejezések száma. Néhány esetben pedig interaktív, pl. nem azt mondja, hogy a Start menüből nyissunk egy DOS ablakot, hanem a segítőszövegben van egy kis gomb, arra kell rákattintani a DOS ablak megnyitásához. Kattintunk, és tényleg.

Új kellékek

Nemcsak az MS Windows, hanem a hozzá tartozó kellékek többsége is megújult, mindegyik alapos kipróbálásához még nem volt elég időnk, csak összefoglalóan sorolhatjuk fel őket. A Paintbrush helyét a Paint vette át, többféle BMP formát kezel, de PCX-et már nem. A Write helyett a WordPadet kapjuk, ami elődjénél sokkal komfortosabb. Két fokozatban nagyító oldalmegjelenítője van (Preview) és Write formátumon kívül ANSI, RTF és Winword 6.0 formátummal is dolgozik. A WordPad mellett megmaradt a korábbi Jegyzettömb.

Multimédia címszó alatt CD-lejátszót, hangerőszabályozást és hangrögzítőt ad a Windows 95.

Hosszú fájlnevek (Na, végre!)

A DOS egyik kényelmetlen öröksége a FAT alapú fájlrendszer a 8+3 betűs fájnevekkel és a fokozatosan szétaprózódó, a merevlemeznek egyre több munkát adó fájlokkal. Az utóbbi problémán a szétaprózódást megszüntető különböző segédprogramok (SpeedDisk, Defrag, Diskopt stb.) rendszeres használata még tud segíteni, de a hosszabb fájlnevekhez mindenképpen új operációs rendszer kellett. A Win95 alatt a fájlnevek maximális hossza 255 karakter, s lehet bennük szóköz, pont, továbbá néhány olyan karakter is, amely eddig nem volt használható. Ugyanez vonatkozik a könyvtárnevekre is. Nem túl szigorú korlát, hogy a teljes fájlazonosító, vagyis a fájlnév az elérési útvonal (path) feltüntetésével együtt legfeljebb 260 karakter lehet.

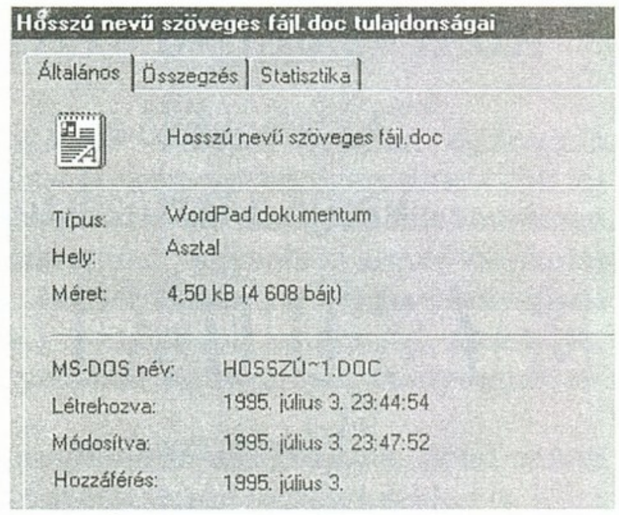
A hosszú fájlnevek kezelésének megoldása egyáltalán nem volt egyszerű, hiszen az eredeti FAT rendszerben a könyvtárbejegyzéshez egyszerűen nincs hely több karakter számára. A teljesen új könyvtárszerkezet viszont alapvetően sértené a kompatibilitást, és rengeteg korábban készült szoftver egyszeriben használhatatlanná válna, vagy adatvesztés következne be. Ezért a Win95 a kompatibilitást szem előtt tartva ugyanazt a — kötet méretétől függően — 12 vagy 16 bites FAT szerkezetet használja, mint az eddigi DOS-változatok.

Ebben a rendszerben egy könyvtárbejegyzés 32 bájtot igényel. A Win95 a hosszú fájlnevekhez egyszerűen annyi ilyen 32 bájtos blokkot foglal le, amennyi szükséges. A kompatibilitás miatt a hosszú névből képez egy, a 8+3-as szabálynak megfelelő rövid nevet (MS-DOS-névnek hívja), és ez kerül az utolsó blokkba, a korábbi programok csak ezt látják. Azt, hogy az előző blokkok tartalmazzák a valódi hosszú nevet, egy eddig nem használt attribútumértékkel jelzi. (A DOS, mint az közismert, négy attribútumot használ, a könyvtárbejegyzésben azonban egy bájt szolgál az attribútumtárolásra, így vannak eddig nem használt kombinációk is.)

A több blokkból álló bejegyzés a hosszú néven kívül más szolgáltatásokat is lehetővé tesz. Az eddigi „létrehozva vagy módosítva” dátum helyett külön tárolja a létrehozás, az utolsó módosítás és az utolsó hozzáférés dátumát is. A fájlok átnevezéséhez elegendő az Alt billentyűvel a fájlhoz tartozó ikonra kattintani, és alatta a fájlnév egyszerűen átírható.

A hosszú nevű fájlokat a korábbi programokkal másolva a teljes név természetesen elvész, csak az MS-DOS név marad meg, adatvesztés azonban nem történik. A hosszú nevek átmeneti elvesztése miatt sok jó segédprogram, mint például a Norton Commander átmenetileg kiesik. Ennek enyhítésére a Win95 tartalmaz egy töredékmentesítőt és a ScanDisk új változatát. Az alatta futó MS-DOS 7.0-t felkészítették a hosszú nevekre.

Ahogy a 14. oldalon lévő képernyőábrán látszik, a DIR parancs a teljes nevet is mutatja. Többi parancsai (például XCOPY) nem veszítik el a hosszú nevet a fájlműveleteknél.



Új filozófia

Mint az Intéző ablakában is látható, a Win95 következetesen foldereknek (mappának) nevezi az alkönyvtárakat. A szóhasználat mögött elvi megfontolások állnak. A Microsoft szerint a felhasználók nagyobb(?) része nem programokkal akar dolgozni, hanem adatokkal, dokumentumokkal.

Eddig elindítottuk az alkalmazást, majd az alkalmazáson belül kerestük meg és töltöttük be a vele kezelhető dokumentumokat. Most közvetlenül a dokumentumra kattinthatunk, és a Win95 annak formátumából (vagy a fájl kiterjesztéséből?) kitalálja a hozzá illő alkalmazást, és automatikusan elindítja azt.

Az installáláskor saját könyvtára alatt létrehoz egy Desktop (asztal) nevű könyvtárat. Minden, ebbe a könyvtárba kerülő adatfájl vagy könyvtár ikonja automatikusan megjelenik a képernyőn, az asztalfelületen, vagyis gyorsan elérhető.

Ugyanez fordítva is működik, a Win95 kellékei által kezelt formátumban (BMP, RTF, Word 6.0 stb.) egyszerűen, az alkalmazás külön indítása nélkül hozhatunk létre dokumentumot az asztalon, amit azután a program fizikailag az asztal könyvtárában tárol.

Egyelőre ennyit az új Windows bemutatásaként; de a további részekre, mint például a plug & play hardverkezelésre, a hálózati lehetőségre, vagy a hordozható gépek támogatására később számainkban visszatérünk.

Csórián Sándor

SYSGUARD

ADATVÉDELMI RENDSZER

Az információ érték!

Védje a fertőzéstől

SYSDOKI v 5

Védelmi rendszer
a vírusfertőzések felderítésére
és a fertőzött programok megtisztítására

- ismeretlen vírusok ellen is használható
- memóriában talált vírusok hatástalanítása
- gyors, egyszerűen kezelhető automatikus ellenőrzés

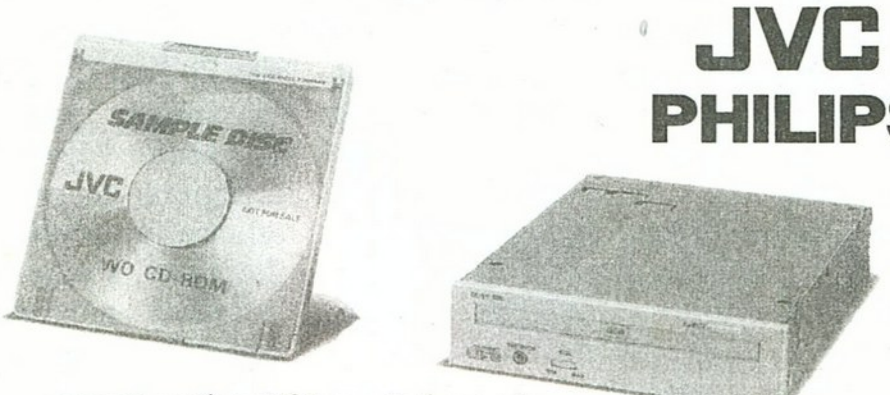
ÆRUS COMPUTER

Aerus Kft.
1076 Bp., Sajó u. 2.
Tel.: 322-4037 , 322-4013
Fax: 116-7089

INFORMÁCIÓKÉRÉS: 0821 ▲

A NAGYOKOS

JVC
PHILIPS



- **mert méretéhez képest rengeteget tud,**
- **mert ezzel sohasem kérkedik és megfizethető.**

Ez a:

CD-R

CD - rekorder és lemez
CD - drive, - jukebox, - torony
Hálózati illesztés

PROCOMP **Procomp-Hungary Kft.**
1107 Budapest, Szállás u. 21.
Tel.: 262-6631, 2618235, 260-4348* Fax: 260-6318

PROCOMP: MINDENT TUD AZ ADATOKRÓL

INFORMÁCIÓKÉRÉS: 0824 ▲

NetWare Connect

egy ideális kapcsolat

- *Segítségével elérhetővé válnak hálózati erőforrásai távoli PC, Macintosh vagy TCP/IP Client felhasználók számára is*
- *Ön is egyszerűen hozzáférhet külső szolgáltatásokhoz (CompuServe, BBS-ek aszinkron minicomputerek)*
- *Megoszthatja kommunikációs HW/SWeszközeit a munkaállomások között, ami jelentős költségmegtakarítást jelenthet*

Vásároljon jogtiszta, hozzáértő forrásból, keresse viszonteladóinkat

WALTON NETWORKING LTD.

Walton Networking Kft.
H-1077 Budapest, Almássy tér 2.
Tel.: 267-9010 Fax: 267-9011
Szegedi Iroda: 6723 Szeged, Sándor u.1.
Tel/Fax: (62) 490-424

NOVELL

A tradícionális disztribútor

INFORMÁCIÓKÉRÉS: 0834 ▲

Újdonságokról — dióhéjban

RAM Doubler for Windows

Gyártó: Connectix Corporation.
Típus: Windows memóriamenedzser.
Becsült ár: 12 000 Ft + áfa.

Ebből a kevesebb mint 70 KB méretű windowsos virtuális meghajtóból bizony a nagy Microsoft is tanulhatna. A program nem tesz mást, mint rugalmasan megnöveli a Windows resource (erőforrás, munkaterület) méretét a duplájára. Mint tudjuk, az alapállapotban mindössze 64 KB-os munkaterület fedezi (többek között) a Windows grafikus felülete elemeinek bejegyzését (ikonok, menük, ablakok stb...). Minél többet használunk ezekből, annál inkább csökken a rendelkezésre álló szabad munkaterület.

Ezen segít a RAM Doubler. Kipróbáltuk. 16 MB RAM-mal ellátott gépünkön elindítottunk egy DOS-taskot, a Microsoft Word for Windows 6-ot, ebben megnyitottunk egy egylapos állományt, amely többoszlopos táblázatot tartalmazott, ebből OLE2-vel beléptünk a Microsoft Excel 5-be, ahol is beemeltünk egy mintegy 70 KB-os, meglehetősen bonyolult felépítésű munkatáblát. Legvégül kértünk az F1-re egy Excel Helpet. Mindezek után a Program Manager Helpben az About ablakra kapcsolva még mindig maradt 50% szabad munkaterület és 21 MB virtuális memória (a kiindulás 24 MB, 80% volt MS Mail MAPI Spoolerrel megspékelve MS Windows for Workgroups 3.11-ben). Ezek után a RAM Doublert üzembe helyezve a fentieket végrehajtottuk, és a végeredmény meggyőzően hatott: 23 MB memória és 55% szabad munkaterület.

A program egyébként saját maga reklámjaként mindig feltünteti azt is, hogy mennyi az annyi vele — és mennyi nélküle. A Windows indításakor, ha az Esc gombot folyamatosan lenyomva tartjuk, akkor a RAM Doubler nem töltődik be. De programból is uninstallálható. Az általunk elindított Windows-alkalmazások számára tökéletesen átlátszó, azaz nem okozott kellemetlen meglepetéseket. QEMM-mel, Netroom-mal, 386MAX memóriamenedzser szoftverekkel problémamentesen összedolgozik.

Ha csak arra jó, hogy elkerülhessük az „Insufficient Memory", „Memory full" és egyéb memóriafaló Windowsos alkalmazások okozta lefagyásokat, máris sok bosszúságtól kíméltük meg magunkat. Ráadásul Windowsunk is stabilabb! Ha rövid a kardod, toldd meg egy lépéssel! Egyébként RAM-ba dőlhet a munkád.

Labels Unlimited for Windows

Gyártó: Softkey.
Típus: Univerzális címkekészítő.
Minimális/ajánlott hardverkövetelmény:
386 vagy jobb processzor, 2/4 MB RAM,
2,5/5 MB szabad merevlemezterület.
Becsült ár: 10 000 Ft + áfa.

Mire való a számítógép? Például gyakran előforduló műveletek automatizálására, könnyebbé, kényelmesebbé tételére. Ezt kiválóan sikerült megvalósítani a Labels Unlimited for Windows (LUW) szerzőinek, ugyanis ki szeret vacakolni körlevélkészítéskor a borítékcímzéssel, vagy szabványos (EAN-8, EAN-13 stb.) vonalkódot tartalmazó árcédulák, névjegykártyák, floppycímkék tervezésével és nyomtatásával?

Mivel windowsos programmal van dolgunk, gond nélkül írhatunk teljes magyar ékezetes karakterkészletekkel is. Kész dBase vagy ASCII szövegállományokból vehetünk át adatokat, címeket. Az importálható grafikus formátumok a következők: TIF, PCX, BMP, EPS, WMF, GIF, TGA vagy PCT.

Több mint 350 féle gyári méret (angolszász és metrikus) közvetlen kezelése, többszáz clipart ábrácska, elforgatott (akár függőleges) írásmód, automatikus sorszámozás.

TrueSpace for Windows

Gyártó: Caligari Corporation.
Típus: Professzionális 3D grafika- és animációkészítő.
Minimális/ajánlott hardverkövetelmény:
386 vagy jobb processzor
és matematikai társprocesszor,
4/8 MB RAM,
8 MB szabad merevlemezterület.
Becsült ár: 88 000 Ft + áfa.

Szavakban nehezen leírható, intuitív képességekkel felruházott VALÓDI háromdimenziós rajzoló és szerkesztő program. Egyformán nélkülözhetetlen segédeszköze a művészi grafikusnak, a belsőépítésznek, a multimédiás prezentációkészítőnek vagy a látványos videoeffektusokkal operáló filmesnek. Testek felületét közvetlenül színezhetjük, mázolhatjuk, festhetjük. Minden azonnal megmozdítható, tetszés szerint bármely irányból körüljárható, akár több fényforrással megvilágítható. Ízelítő a választható anyagmintákból: polírozott fém, átlátszó vagy áttetsző üveg, matt gipsz, márvány... meg ami jólesik. Kiválasztott trajektórián, azaz tetszés szerinti girbegurba pálya mentén összetett testek deformálása, mozgatása, animálása, (a kameránk is mozoghat, nincs fix helyhez rögzítve!).

Támogatott formátumok oda-vissza: DXF, 3D Studio, BMP, TGA, BMP, AVI.

Egyéb nyalánkságok: vizuális idő editor több objektum szinkronizált mozgatásához, ray-trace modul, 8000 x 8000 pixel max. felbontású rendering, truecolor (24 bites) imagekezelés, természetes effektusok, például köd. Mindez egy jól kézhez álló vezérlőfelületen, amellyel már egy 6 éves srác is azonnal elboldogul.

ABC Flowcharter 4.0 for Windows

Gyártó: Micrografx, Inc.
Típus: Professzionális folyamatábrakészítő.
Minimális/ajánlott hardverkövetelmény: 386
vagy jobb processzor és matematikai

társprocesszor, 4 MB RAM, 5/21 MB szabad merevlemezterület, CD-olvasó az ajándékba kapott betűkészlet (font) CD-hez.
Becsült ár: 64 000 Ft + áfa,
upgrade/competitive upgrade:
26 000 Ft + áfa.

Ez már egy „megaköteg". Eleve 4 különálló modult tartalmaz:

1. ABC FlowCharter, azaz maga a névadó.
2. ABC Data Analyzer: gyártási folyamattervező és minőségellenőrzési folyamatábrák a kritikus út meghatározásához és a gyártás időbeni nyomonkövetéséhez.
3. ABC SnapGraphics: előre elkészített diagramminták kényelmes és lusta menedzserek számára (Flow, Network, Freestyle, Org, Timeline, Comparison, Circle/Spoke, Netmap, Connection, Venn, Pyramid, Checklist, Tree stb.)
4. ABC Viewer: díjtalanul felhasználható (royalty free) ábramegjelenítő.

Új statisztikai diagramok: Pareto, hisztogram, Scatter stb. Adatmezők, angol helyesírás-ellenőrzés, buborék formájú magyarázómezők, új impresszív kezelőfelület (bubble help, hint ribbon, rengeteg munkagyorsító nyomógomb), automatikus vonaltalálkozások, több mint 750 előre elkészített rajzelem, doboz, forma, jel, piktogram.

Más konkurens termékről (például Visio, CorelFlow, AllClear, EasyFlow stb.) kevesebb, mint fele árért csábulhat át az ember!

Microsoft Flight Simulator 5.1 CD Edition

Gyártó: Microsoft Corp.
Típus: professzionális repülésszimulátor, amelyikben nem lőnek.
Minimális hardverkövetelmény:
386SX vagy jobb processzor, 4 MB RAM, 20 MB szabad merevlemezterület, CD-olvasó, SVGA kártya ajánlott.
Becsült ár: 10 000 Ft + áfa.

A már több mint fél éve várt FS 5.1 CD végre megérkezett. A lemezes változatot hónapok óta nyúztuk, de ebben az egész földgömböt körülröpdöshetjük, fotórealisztikus SVGA tájakat, valamint 250 repülőteret érinthetünk. Háromdimenziós felhőkbe szállhatunk bele, közben villám csapkod körülöttünk. A nappal és éjszaka váltakozásainak megfelelően este kigyúlnak a városok fényei... Tavasszal zöldbe borul a természet (természetesen az északi földtekén a kontinentális éghajlatú részeken). A lezuhanást is „szikronizálta" a Microsoft: élethű hanghatások közepette találkozhatunk az anyafölddel.

10UJJ Gépírásoktató és Gyakorló Programcsomag

Gyártó: Hunit Oktatásszervező és Szolgáltató Bt.
Típus: tízujjas vakgépírás-oktató.
Változat: magyar, alap, 1.30.
Minimális hardverkövetelmény:
80286 processzor.
Ára: telepítésszámtól függően
1050–3200 Ft/példány + áfa.

A program a számítógép billentyűzetén történő beírási műveletek leghatékonyabb módjának, a tízujjas vakgépírás technikájának elsajátítását teszi lehetővé. Kezelése egyszerű. Lehetne akár első PC-s programunk is. Használatához nem kell semmiféle segédanyag, minden szükséges oktatási, gyakorlási segítséget megkapunk közvetlenül a képernyőről. Még olyan tanácsokkal is ellát bennünket, mint az üléshelyzet, kéztartás, tanulási menet stb. Begyakoroltatja a billentyűtábla kezelését. Sok feladatot ad a gépíráshoz. A feladatoknál azt és annyit gépelhetünk, amit és amennyit akarunk. Hibamutató jelzi, hogy hol rontottunk. Eredményeinket nyilvántartja, azokat bármikor megnézhetjük. Beépített metronóm segíti az ütemes írás elsajátítását, és még számos praktikus ötlet támogatja a hatékony tanulást. A tízujjas vakgépírás elsajátítása után különösen szövegszerkesztőnk használatában érezhetjük teljesítményünk ugrásszerű javulását.

Eredményesen használható már az általános (!) és középiskolákban is. A felnőttképzésben, a munkahelyeken pedig az adminisztrációs munka jelentős felgyorsítását lehet vele elérni.

A fent bemutatott szoftverek az Alaplap Postán keresztül is megrendelhetők. (Válaszlevelezőlap a középső kartonon.)

Pro/Junior: lehetőség jó áron

Amikor a közepes az optimális!

A piaci elemzések egyértelműen azt mutatják, hogy egyre növekszik az igény a *high-end* és a *low-end* kategória között fennálló űr betöltésére. Erre a kihívásra választ adva napjainban egy új nagyságrend, az ún. *midrange* (középkategória) van kialakulóban, ahol egy munkahely költsége a 15-30 ezer dolláros tartományba esik. Szakértők ennek a piacnak a nagyságát a világon 600 ezer — 1 millió (!) munkahelyre taksálják. Így az ipar jelentős szegmensében számíthat sikerre egy jól kialakított megoldás.

A CAD/CAM piacot új tendenciák megjelenése jellemzi. A mainframe gépekre fejlesztett szoftverek mellett piacra kerültek a már munkaállomásokon futó, korszerűbb technológiákat felvonultató szoftverek újabb generációi. Ezekből formálódott meg az ún. high-end (felső) kategória, amelyben teret nyert előbb a háromdimenziós felületmodellezés, majd a testmodellezés. A szoftvergyártók a komplex, bonyolult feladatok megoldását kínálják ezeknél a rendszereknél. Az egy munkahelyre jutó hardver- és szoftverköltség együtt átlagosan 40-80 ezer dollárt tesz ki.

A másik nagy csoportot az ún. low-end (alsó) kategóriás szoftverek alkotják. Ebbe a táborba alapvetően az olcsó, kétdimenziós rajzolószoftverek tartoznak. Itt egy komplett munkahely 5-10 ezer dolláros költséggel alakítható ki.

Hosszú távon a midrange szoftverek valószínűleg jelentősen szűkítik a kétdimenziós rajzolószoftverek játékterét, a csúcsrendszerek létjogosultsága azonban továbbra is megmarad a különösen komplex feladatok megoldására.

Vizsgáljuk meg, mi az, amit elvárhatunk egy midrange szoftvertől?

Szükséges és elégséges

— Örökölje a high-end rendszerek modellezőképességeinek jó részét.

— Olvassza magába a low-end szoftverek rajzkészítő tudását.

— Gyors, hatékony szoftverműködést biztosítson komplex feladatok esetén is.

— Legyen egyszerűen kezelhető, könnyen tanulható.

— Viszonylag alacsony legyen a kezdő ár.

— Elvárható még a bővíthetőség, a 100%-os kompatibilitás a high-end rendszerek irányába (közvetlen adatátviteli lehetőség, adatkonverzió IGES, DXF stb. nélkül; a high-end rendszerrel megegyező menüszerkezet stb.).

A Parametric Technology Corporation (PTC) a Pro/Junior 1995. januári bejelentésével elsőként jelent meg a piacon olyan szoftverrel, amely minden tekintetben kielégíti a fenti feltételeket. Kövessük végig, hogy milyen módon valósulnak meg a felsorolt elvek a gyakorlatban a Pro/Junior esetében:

— A Pro/Junior a Pro/Engineer 14 változaton keresztül optimalizált, parametrikus, alaksajátosságokon alapuló („feature based"), a maga nemében egyedülálló testmodellező technológiát követ.

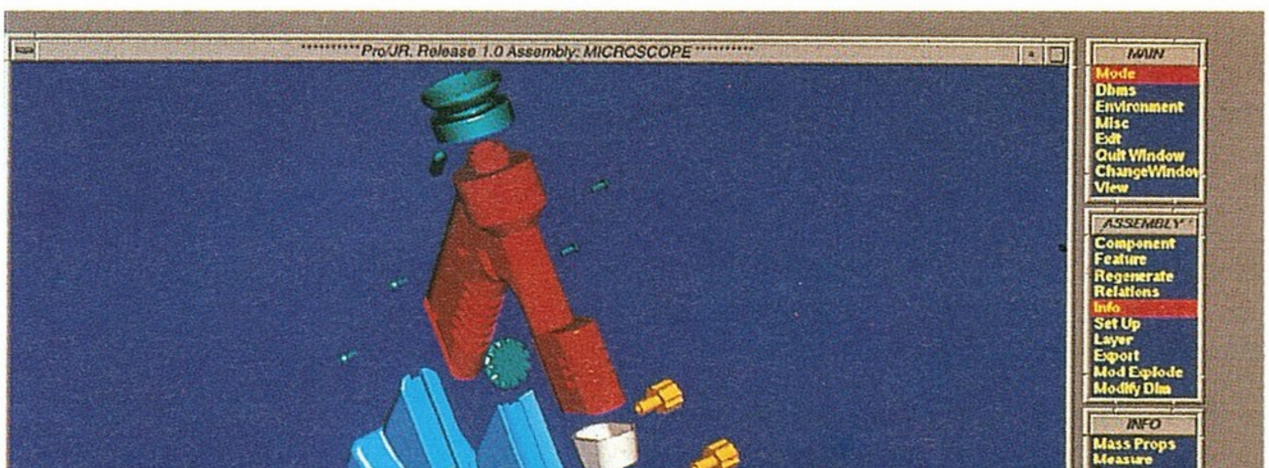
— Az elkészített testmodellekről ANSI, ISO, DIN, JIS szabványú műhelyrajzok automatikusan készíthetők.

— A tömör, C-ben írt szoftverkód a szerényebb képességű munkahelyeken is hatékony munkakörnyezetet biztosít.

— A Pro/Junior menüszerkezete áttekinthető, a menü intelligens automatizmusokat tartalmaz, egyszerűen makrózható. A szoftver használata közben bármikor kérhetünk segítséget. Az online help grafikus felülete egyértelmű tanácsokat szolgáltat a felhasználóknak. A betanulást oktatóprogramok teszik gyorsabbá.

— A sebességre optimalizált szoftver futtatható az olcsó Unix munkaállomásokon, valamint Windows NT alatt PC-n. Így minden felhasználó kiválaszthatja a pénztárcájának leginkább megfelelő rendszert. Egy teljesen kiépített munkahely (hardver + szoftver) ára 16-18 ezer dollárnál kezdődik.

— A Pro/Juniorban elkészített modellek, rajzok konverzió nélkül közvetlenül beolvashatók a Pro/Engineer rendszer más szoftvereibe. A Pro/Junior szabadon upgrade-elhető a Pro/Engineerre. Ha a felhasználó elsajátította a



szoftver kezelését, azzal a Pro/Engineer használatát is megtanulta, mivel a menük felépítése megegyezik. Így upgrade esetén nemcsak az elkészített modellek, rajzok vihetők át gond nélkül a Pro/Engineer alá, hanem a szoftver tanulásába fektetett munka sem vész kárba.

Miért lesz megfelelő?

A Pro/Junior fontosabb tulajdonságai, képességei részletesebben az alábbi pontokban foglalhatók össze.

Jól használható tervezési alaksajátosságok

A Pro/Junior hathatós alaksajátosságokat nyújt, ezek létrejöhetnek egyrészt vázlat alapján, másrészt ún. „pick and place" módon. Az előbbinél profil kihúzása, körbeforgatása, vezérgörbén történő végigsöprése, illetve profil(ok) egymásba mosása által képzett térfogatról (létrehozásról vagy elvételről) van szó. Az utóbbinál héj, formázási ferdeség, változó és állandó sugarú lekerekítések, élletörés, stb. létrehozására gondoljunk.

Alaksajátosság-műveletek

Erőteljes alaksajátosság-műveletek csoportja áll rendelkezésre. Lehetőség van a vázolt profil megváltoztatására, alaksajátosság törlésére, átmeneti letiltására, aktiválására, ismétlődő elemek képzésére, az alaksajátosságok sorrendjének utólagos megváltoztatására.

Parametrikus kapcsolatok

A tervezői szándék megvalósítható az alkatrész és/vagy az összeállítás paraméterei közötti kapcsolatok felállításával. Egy paraméteren történő változtatás hatása a felállított kapcsolatoknak megfelelően a tervezésben mindenütt megjelenik.

Testjellemzők számítása

A Pro/Junior automatikusan számítja ki az összes lényeges testjellemzőt (tömeg, felszín, súlypont stb.) mind egyedi alkatrészekre, mind összeállításokra.

Táblázatvezérelt alkatrészek

Alkatrészcsaládok hozhatók létre egyetlen ún. kiindulási modellből, egyszerűen a kiválasztott méretek hozzáadásával egy alkalmas táblázathoz.

Kényszerek parametrikus elhelyezése

Az összeállítás alkotóelemei parametrikus kényszerek segítségével szerelhetők össze (mint például a felületek egymásra fektetése, egymáshoz illesztése, csap elhelyezése furatban stb.). Az így elkészített összeállításokkal mozgástanulmányok végezhetők.

Részleges és teljes ütközés- és tűrésvizsgálat

A Pro/Junior automatikusan vizsgálja az ütközést a kiválasztott felületek, alkatrészek és részösszeállítások között, kiszámolja a közös térfogatot, és grafikusan is megmutatja, hogy hol van az átfedés. Az alkotóelemeket vizsgálni lehet a maximális és minimális tűrésértéküknél, eszerint számolva az ütközéseket, illesztéseket és a testjellemzőket.

Alkatrészek módosítása

Az alkatrészek méreteit közvetlenül az összeállításban is módosítani lehet.

Automatikus darabjegyzék-készítés

A darabjegyzék közvetlenül az összeállításból származik. Az eredmény elmenthető egy fájlba, amit később fel lehet használni a rajzkészítésnél vagy közvetlenül más alkalmazásokban is.

Teljes mértékben asszociatív műszaki rajz

Asszociatív rajzok készíthetők közvetlenül a modellből. Bármely változtatás a rajzon magáa után vonja a modell és az összeállítás módosulását, és ez fordítva is igaz.

Az ANSI, ISO, DIN, JIS szabványokkal való együttműködés

Különböző ipari vagy vállalati rajzdokumentáció-szabványnak megfelelő rajzok készíthetők.

Asszociatív rajzi nézetek

A Pro/Junior a nézetek, metszetek teljes skáláját származtatja a modellből (normál, vetített, rész- és törtnézet; sík, lépcsős, kitörés stb.). A méretek és más kapcsolódó információk automatikusan megjelennek a nézeteken.

Kiegészítő méret, tűrés és egyéb

A rajzra tetszőleges kiegészítő, tájékoztató méretek helyezhetők el. Helyzet- és alaktűrés, valamint érdességi jel szimbólumkönyvtárak állnak rendelkezésre.

Interaktív takartvonalas megjelenítés

A felhasználó interaktív módon tud dolgozni a takartvonalas megjelenítéssel. A takartvonalas megjelenítésnél beállítható, hogy a nem látható vonalak más színűek legyenek, vagy egyáltalán ne látszódjanak. A takart vonalakat és felületeket a nézet orientációjának módosítása nélkül ki lehet választani.

Árnyékolt megjelenítés

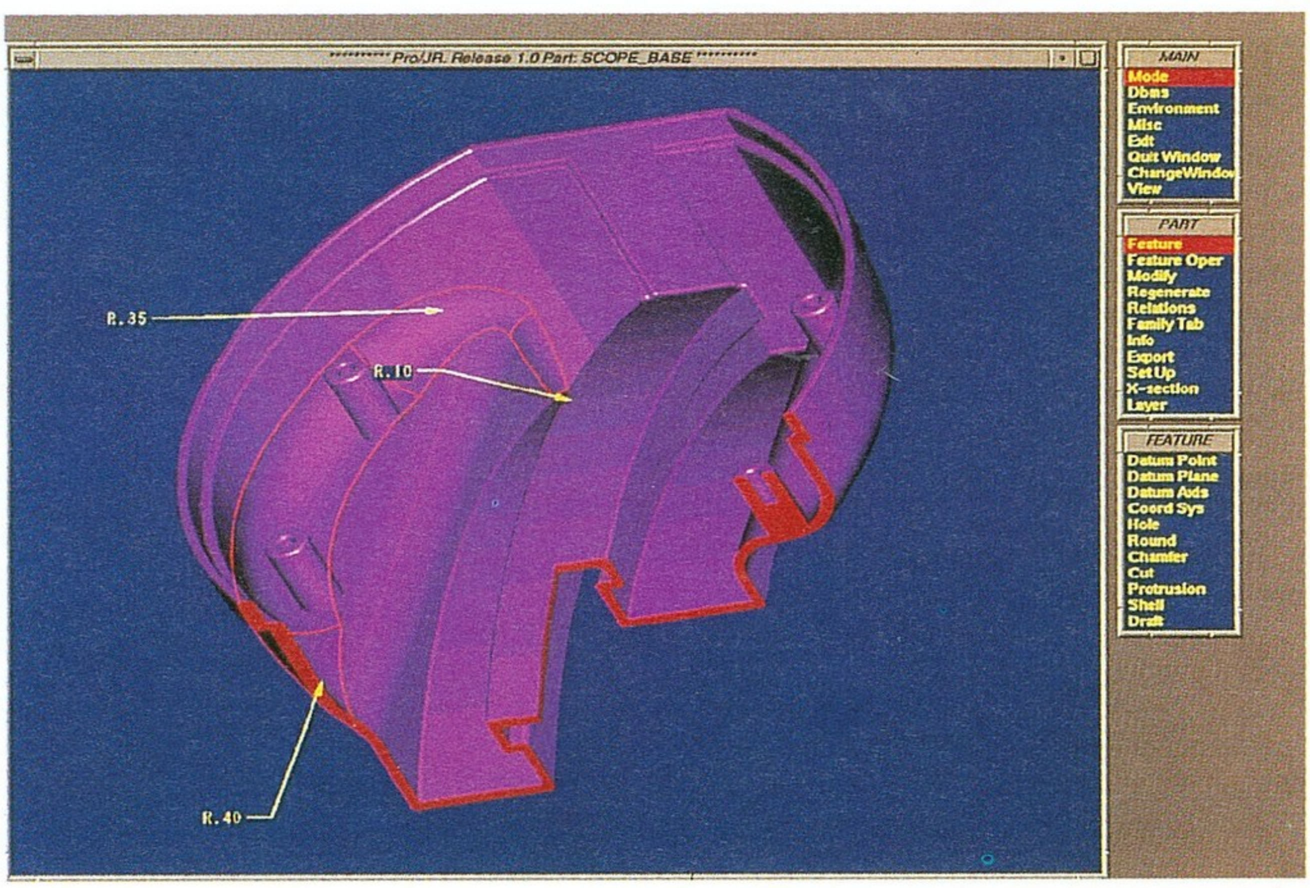
Az alkatrészek és összeállítások a jobb láthatóság kedvéért árnyékolt módban is megjeleníthetők. Lehetőség van mozgatásra, nagyításra, forgatásra.

Összeállításban robbantott nézet

Az összeállítási modell és rajz szétrobbantható a nem látható alkotók vizsgálatához.

Standard import/export interfész

A Pro/Junior szabványos interfészformátumokon (IGES, DXF, SET, SLA,



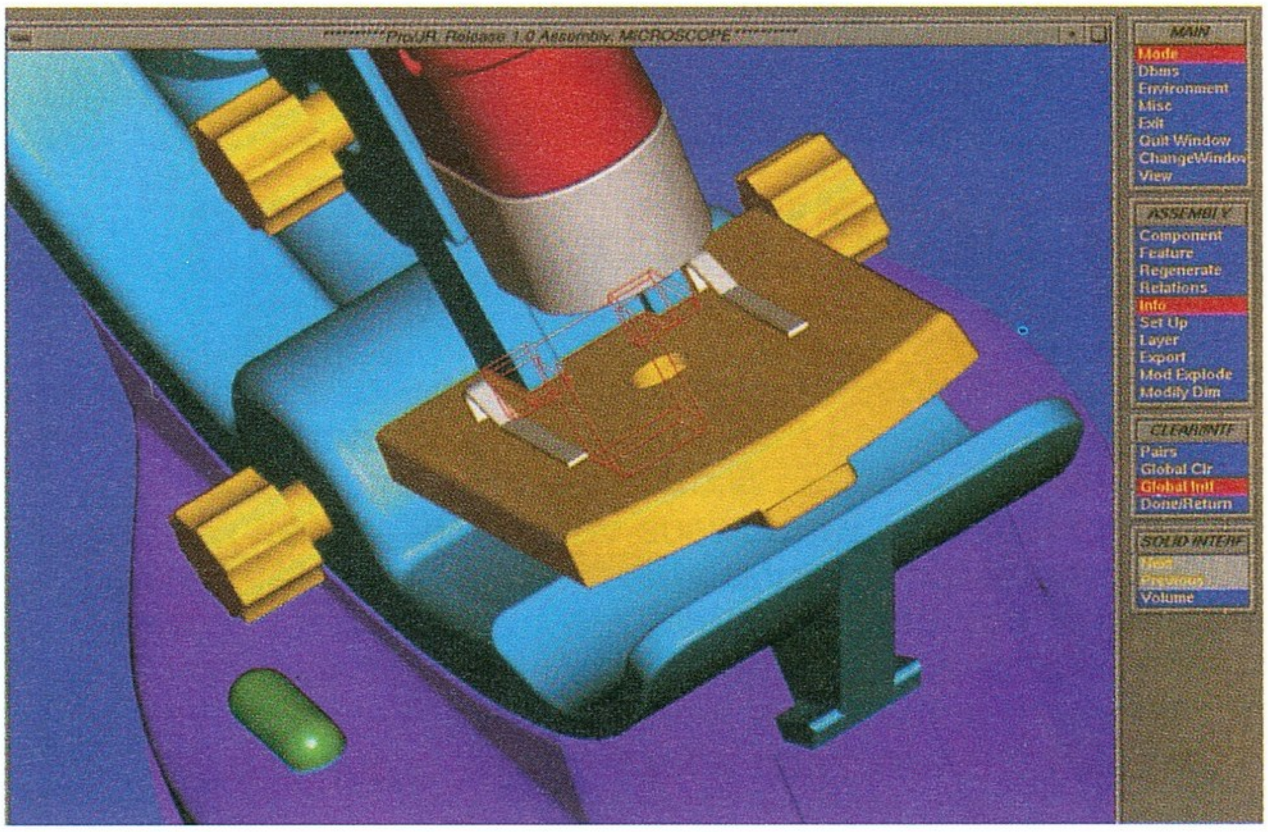
Render és Inventor) keresztül kommunikál más CAD rendszerekkel.

Széles körű plottertámogatás

A Pro/Junior támogatja az iparban szabványosnak számító plotterformátumokat (PostScriptR, Color PostScript, HPGL, HPGL2, Versatec, Calcomp és Gerbert fotoplotter).

„Paraméterek"

A PTC, a Pro/Junior fejlesztője 1988-ban a felhasználói igényekre érzékenyen reagálva megalkotta az első parametrikus, alaksajátosságon alapuló modellezést megvalósító CAD/CAM rendszert, a Pro/Engineert. Hasonló megoldással azóta több cég is megjelent a piacon: igaz, újra kellett fogalmazniuk termékeiket, és a Boole-algebrán alapuló modellezésről való áttérés komoly áldozatokat igényelt a CAD/CAM cégektől.



Hogy a PTC, amely 1988-ban egyszer már piacformáló tényezővé vált a high-end piacon a Pro/Engineer révén, hasonló sikert ér-e el a midrange-piacon a Pro/Juniorral is, rövidesen kiderül: az ipar el fogja dönteni az új termékek, illetve a kategória sorsát. Bizonyos azonban, hogy a szoftver a magyar piac speciális igényeihez jól illeszkedik hatékonyságával, kedvező árával.

A Pro/Junior izgalmas, új világba kalauzolhatja el az igényes, piacgazdasági kihívásokkal szembenéző, szerényebb anyagi lehetőségekkel rendelkező magyar felhasználókat. A PTC hivatalos magyarországi disztribútora mindenesetre már több mint 100 Pro/Engineer-munkahelyet installált.

Nyirő Ferenc—Vadász Gábor

Elektronikus laboratórium PC-n

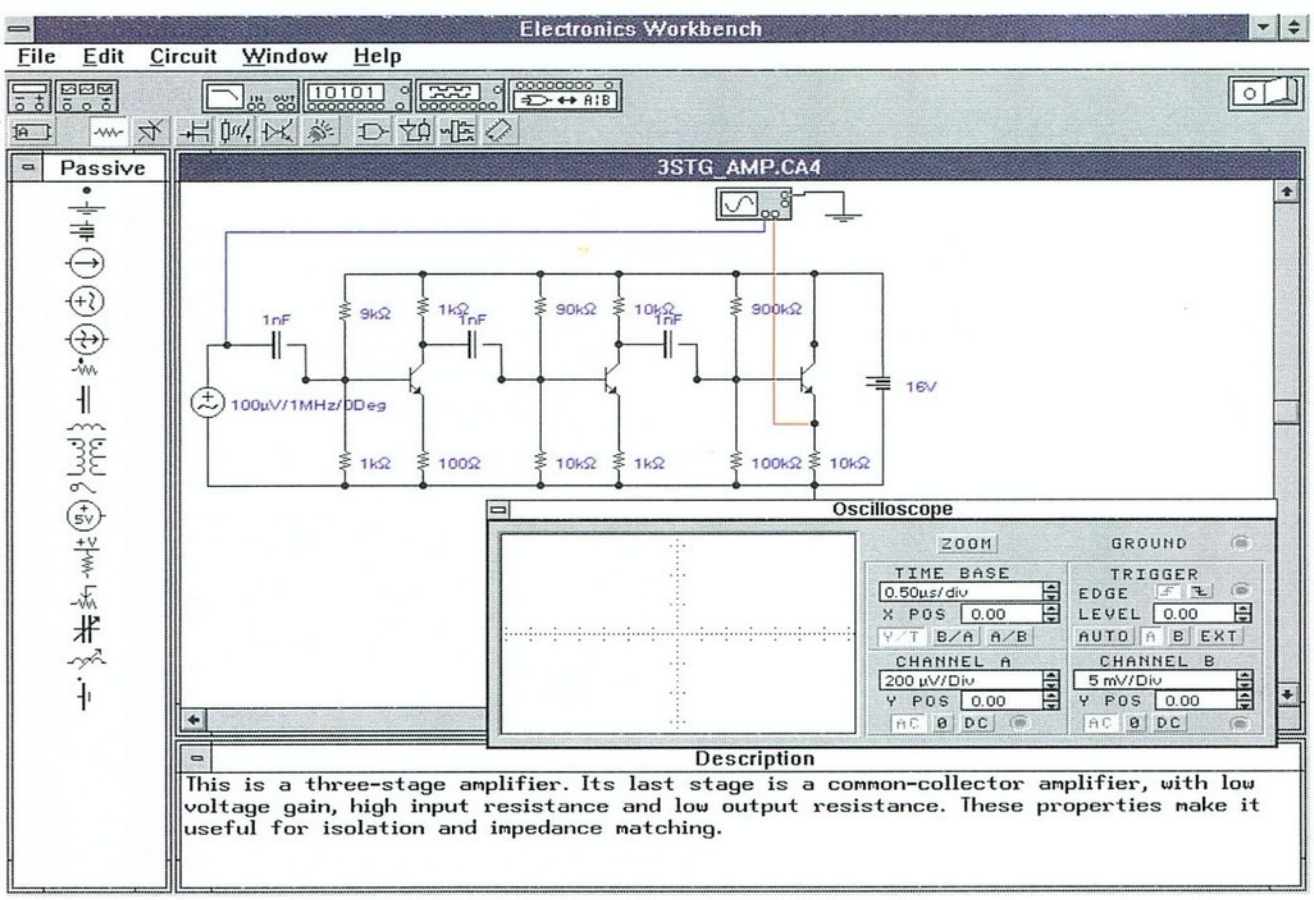
Egy program „iskolás” szemmel

Egy kanadai illetőségű szoftvergyártó cég — az Interactive Image Technologies Ltd. — Electronics Workbench nevű programja létezik DOS, Windows és Macintosh alatt futó változatokban is. A windowsos változat elindítása után egy valóságosnak tűnő laboratóriumi munkaasztal tárul elénk; megjelenik egy üres munkafelület, bal szélén az éppen kijelölt alkatrészcsoport rajzszimbólumait kínálva, fent a polcon pedig sorakoznak a mérőeszközök.
A termék 4.0-s változatát a forgalmazó Sagax jóvoltából a budapesti Corvin Mátyás Gimnázium és Műszaki Középiskola oktatói és diákjai közösen tesztelhették. Tapasztalataikat foglalja össze az alábbi írás.

A „szimuláció” kifejezés szó szerint tettetést, mímelést jelent, lekicsinylő értelmet fejezhet ki a hétköznapi beszédben. A tervezőmunkában azonban igen hasznosnak bizonyult a valóság gépi utánzása, azaz a szimuláció. Nagyfokú előrelátást (hovatovább jóslást) tesz lehetővé.

A bennünket körülvevő, szinte mindent tudó elektronikus gépek túlnyomó többsége már majdnem átláthatatlanul bonyolult integrált áramkör, persze ezeket is emberek tervezték, készítették, ezek is apró építőkövekből állnak. Ha meg akarjuk érteni működésüket, vagy egyszerűbbeket magunk is el akarunk készíteni, ezt megtehetjük a PC-nk előtt ülve, anélkül, hogy meg kellene vásárolnunk a még nem is biztosan specifikált drága alkatrészeket.

Szokás szerint előbb tervet, azaz elvi kapcsolási rajzot szerkesztünk. A nagyon precíz mérnöki tervezést szolgáló profi számítógépes áramkörtervező szoftverek némileg redukált változatai ma már hozzáférhetők és alkalmazhatók kisebb hardverigény teljesülése esetén is, például az iskolákban, vagy akár otthon is. E programok nemcsak a szorosan vett tervezést, hanem a megalkotott elektronikai áramkör ellenőrzését is biztosítják az ún. szimulációs eljárással. Ebből adódóan a szimulációs programok, köztük a szóban forgó is, jó eszközei lehetnek a középfokú szakmai képzésnek.



Az áramkörök kétfélék

Hagyományos felosztás szerint az áramköröknek két csoportja létezik: az analóg (folytonos értéktartományban működő) és a digitális (kétértékű jelekkel működő).

Az analóg jelekkel dolgozó áramkör esetében a szimuláció voltaképpen abban áll, hogy a számítógép a valóságos (és összetett tulajdonságú) alkatrészeket egyszerű helyettesítő modellek összekapcsolásával „utánozza”, és ezek egyenleteinek megoldásával számítja ki a kapcsolás csomópontjainak potenciálját. A valódi alkatrészeket ugyanis kétpólusú alapelemek (R, L, C, feszültségforrás) hálózatával helyettesítve, viszonylag egyszerű összefüggéseket használhatunk (Ohm-, Kirchhoff-törvények) a működés vizsgálatakor, megengedhető elhanyagolások mellett.

A digitális logikai áramkör szimulációs vizsgálata az ún. igazságtáblázat teljesülését ellenőrzi a Boole-algebra elmélete alapján. Ebben az esetben a modellek viselkedése is igazságtáblázatként van leírva.

A programok zöme is kétféle

A lényegesen különböző két leírásmód miatt léteznek külön analóg és külön digitális szimulációs programok. A kevert módú szimulációt csak kevés program tudja szolgáltatni. A vegyes felépítésű kapcsolást például úgy lehet vizsgálni, hogy a digitális modell bemenetén analóg, középen ideális logikai, kimenetén ugyancsak analóg részekre van tagolva (igényesebb esetben e láncba valahol késleltetés is lehet beiktatva). A be- és kimeneti részeket analóg módon kezelik; s a központi logikai részt vizsgálják a Boole-algebra szabályai szerint.

Mi most arra az egy programra térünk rá, amelynek kapcsán kellett mindezt eddig elmondanunk.

A kapcsolási rajz szerkesztése tehát azzal kezdhető, hogy a kiválasztott al-

katrészcsoport rajzszimbólumai közül egyet megragadva az egérrel, bevonszoljuk a munkafelületre, és ott, leendő helyén elengedve már ez az alkatrész rendelkezésre is áll a készülő kapcsolásunkhoz. A program azonnal ajánl egy szokásos értéket a kérdéses alkatrésznek, de ezt természetesen könnyedén meg is változtathatjuk a kétszeres rákattintás után megjelenő ablakban, saját céljainknak megfelelően.

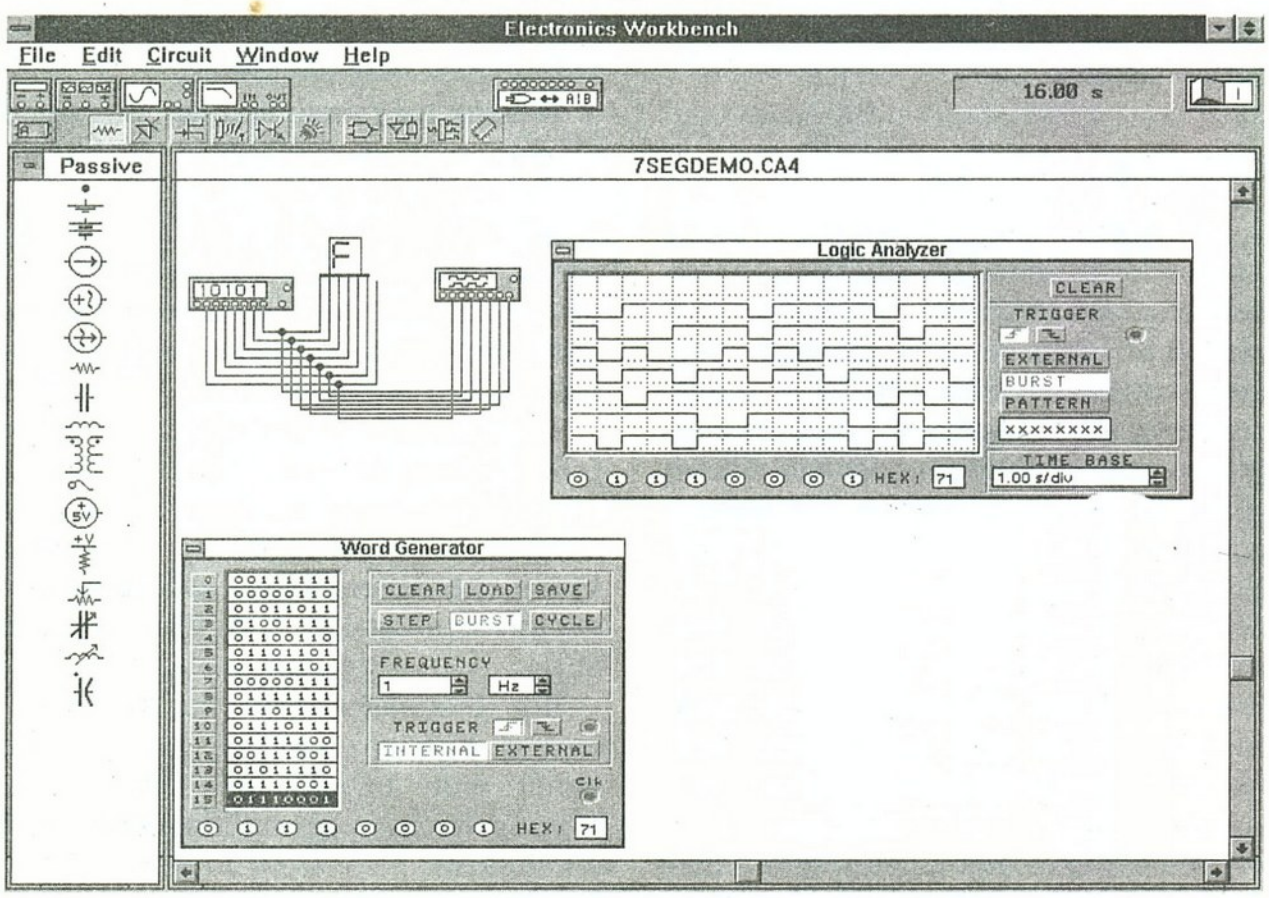
A program ismeri az elektronikában használatos alkatrészek szinte mindegyikét, s ezeket a SPICE (Simulation Program with Integrated Circuit Emphasis) alapú szimulációs programoknál megszokott modelljeikkel használja, sajátságos csoportokba rendezve.

Passzív csoport: ellenállás, potenciométer, kondenzátor, tekercs, transzformátor;

Aktív csoport: egyen- és váltakozó áramú feszültség- és áramforrások, rétegtranzisztorok, műveleti erősítők, FET-k, vezérelt elemek (például feszültségvezérelt áramforrás, relé, kapcsoló), digitális kapuk (IC components), kombinációs digitális diódák (például fél és teljes összeadó), szekvenciális digitális áramkörök (például számlálók), hibrid — analóg és digitális — áramkörök (például A—D vagy D—A átalakító).

Áramkörmustra

A lehelyezett alkatrészek szimbólumainak vonallal való összekötése, azaz a huzalozás után kész az áramköri kapcsolásunk, amely igen sokszor vegyes, analóg és digitális részeket is tartalmazhat.



Vizsgálni (analizálni) a megszerkesztett (vagy tárolóból betöltött) áramkört az alábbiak szerint lehet:

1. Be kell huzalozni az áramköri rajz kérdéses pontjaira a készletből a választható vizsgálóeszközök közül a szükségeseket. Ezek lehetnek:

— Indikátorok (egyszerű állapotmegjelenítők).

— LED.

— Hangjelző (a PC hangszórója).

— 7 szegmenses kijelző.

— Mérőműszerek:

Multiméter; áram-, feszültség-, ellenállás-mérési lehetőséggel.

Hullámforma-generátor; jelalakok: szinusz, impulzus (melynek választható az alakja és a kitöltési tényezője).

Kétsugaras oszcilloszkóp.

Bode-plotter (Bode-diagramrajzoló).

Szógenerátor digitális vizsgálatokhoz.

Logikai analizátor.

Logikai konverter.

Voltmérő.

Ampermérő.

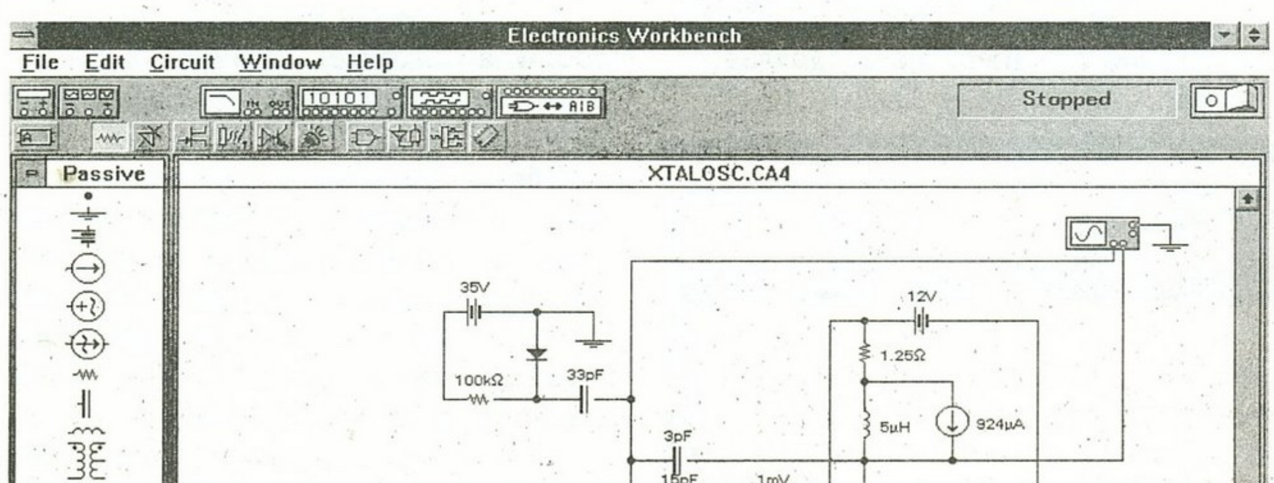
Ez utóbbi kettőből több is beköthető a kapcsolásba egyidejűleg, míg a többi mérőeszközből csak egy-egy lehet hozzákapcsolva az analizálandó áramkörhöz. A műszerek előlaprajzolata jól utánozza a szokásos valódit, és bemeneti-kimeneti ellenállásuk is beállítható a valódiak értékére.

2. Választani lehet az analizálás módjai közül. A lehetséges analizálási módok analóg áramkörökön:

DC-analízis. Mint egy régóta bekapcsolt egyenáramú hálózat csomóponti feszültségeit és ágainak áramait számítja ki a gép.

AC-analízis. Mint egy tisztán szinuszos váltakozófeszültséggel működő hálózat feszültségeit és áramait számítja ki a gép, és megjeleníti az amplitúdó- és fázisviszonyokat.

Tranziens analízis. A feszültségugrás hatására való viselkedés az idő függvényében. Vegyük például a program mintakészletében amúgy is meglevő oszcillátort, amelynek működését a hozzákapcsolt jelalakvizsgáló oszcilloszkóppal ellenőrizzük. Az oszcilloszkópon beállítjuk a megfelelő idő- és feszültségléptéket, a triggert stb., és a munkaasztal jobb felső sarkában bekapcsoljuk a tápfeszültséget. Ez egy ún. egységugrást jelent az áramkörre kapcsolva, és látható, hogy megérkezésének pillanatától számítva csak néhány



periódusnyi idő után kapjuk a csillapítatlan rezgést.

Logikusan...

A logikai vizsgálat a modellválasztáskor beállítható jelszintekkel és késleltetésekkel történhet. Lehetséges modellek az Elektronics Workbench programban: ideális, TTL, CMOS. A kapuknak és a többi logikai elemnek normál (totemoszlopos) és nyitott kollektoros (OC) kimeneti megoldása választható, az L, LS, HC, HCU változatok közül. A logikai kapuk modellje a főbb jellemzőket veszi figyelembe: a magas bemeneti feszültségszintet, az alacsony bemeneti feszültségszintet, a jelterjedési időt, az átváltás feszültségszintjét, valamint az illető logikai egység igazságtáblázatát.

Vizsgáljuk meg a program példái közül például egy hétszegmenses kijelző működését: behívjuk a tárolt kapcsolás rajzát, DC-analizálást kérünk, bekapcsoljuk a „tápegység" kapcsolóját, és a logikai analizátor képernyőjén látható, hogyan változnak a logikai szintek a hét szegmensvezetéken, miközben a bemeneti szógenerátor végigjárja a lehetséges kombinációkat.

Hibrid áramkörök vizsgálata

Válasszuk ki analizálásra például egy analóg—digitális átalakító áramkörét. A modellablakban beállítjuk a referenciafeszültség, az órajel-frekvencia nekünk megfelelő értékét, potenciométert „kapcsolunk" a bemenetre, hétszegmenses megjelenítőt a kimenetre. Bekapcsolva a tápfeszültséget, és változtatva a potenciométer csúszkájának helyzetét (ami százalékban látható), a kimeneti kijelzőn szépen követhető a bemeneti feszültség értékváltozása számjegyes megjelenésben.

Nézzük még meg, hogyan végezhető néhány gyakran előforduló mérési feladat ezzel a programmal!

A transzferkarakterisztika felvétele során vezérelt generátorról működtetjük az elemet (például logikai kaput), indikátorként oszcilloszkópot használhatunk X-Y üzemmódban.

Kondenzátor töltődésének folyamata (ellenálláson át való töltés) láthatóvá tehető úgy, hogy az egységugrásként rákapcsolt feszültséget, és a sorosan kötött mérőellenálláson eső feszültséget az oszcilloszkóp X és Y bemenetére adjuk.

Egy diódakarakterisztika kirajzoltatása a kondenzátoréhoz hasonló, de ennél nem egységugrás-feszültséget, hanem háromszög hullámalakút kell alkalmazni.

A program további szolgáltatásai

1. A teljes áramköri rajzból kivághatunk egy kerettel körülhatárolt részt, amelyet ezután önálló egységnek tekint a program. Ez egy ún. subcircuit (részáramkör), és mint egy „fekete doboz" fog a rajzon szerepelni, aminek belső áramkörét egy saját készítésű modellként definiáljuk.

Így például felvehetünk egy kvarckristályt is az elemek közé, annak ellenére, hogy az alapkészletben nem szerepel ilyen.

2. Az eszközmodellek egy meglévő átalakításával szerkeszthetőek: edit, copy, paste, delete, new library.

3. Különösen értékesek a logikai konverter szolgáltatásai:

— Logikai szimbólumokkal megrajzolt hálózatnak megadja az igazságtáblázatát.

— Igazságtáblázat alapján felírja a Boole-algebrai egyenletet.

— Egyszerűsíti a logikai egyenletet.

— Egyenlet alapján megrajzolja a logikai hálózatot.

— Egyenlet alapján magadja az igazságtáblázatot.

— Egyenlet alapján megrajzolja a NAND-kapukkal megépíthető ekvivalens hálózatot.

4. A programmal adott katalógus terjedelme több száz elem (alaplista a gépkönyvben és a szórólapon), bővítés lehetséges a Model Set Bundle csomaggal.

5. Kinyomtathatók a kapcsolási rajz, az alkatrészlista, az analízis eredményei.

6. Installáláskor választhatunk a program által használatos ANSI vagy DIN szabványú rajzjelek közül.

7. SPICE-formátumok importálási és exportálási lehetősége.

8. Netlist generálása nyomtatottáramkör-tervezők részére.

Meggyőződtünk...

Az eddigiek után még egyszer — „megalapozottan" — jelenthetjük ki, hogy az Electronics Workbench program jól használható a középfokú szakmai képzésben. Mind az elméleti működés tanári magyarázatához (kivetítve a monitor képeit), mind a laboratóriumban a tanulók egyéni munkájához, vagy akár a tanulók hobbimunkájuk tervezéséhez (például a könyvtárban elhelyezett számítógépeken).

Az egérrel szinte minden művelet elvégezhető, emellett vannak azért ún. hotkey-kombinációk is.

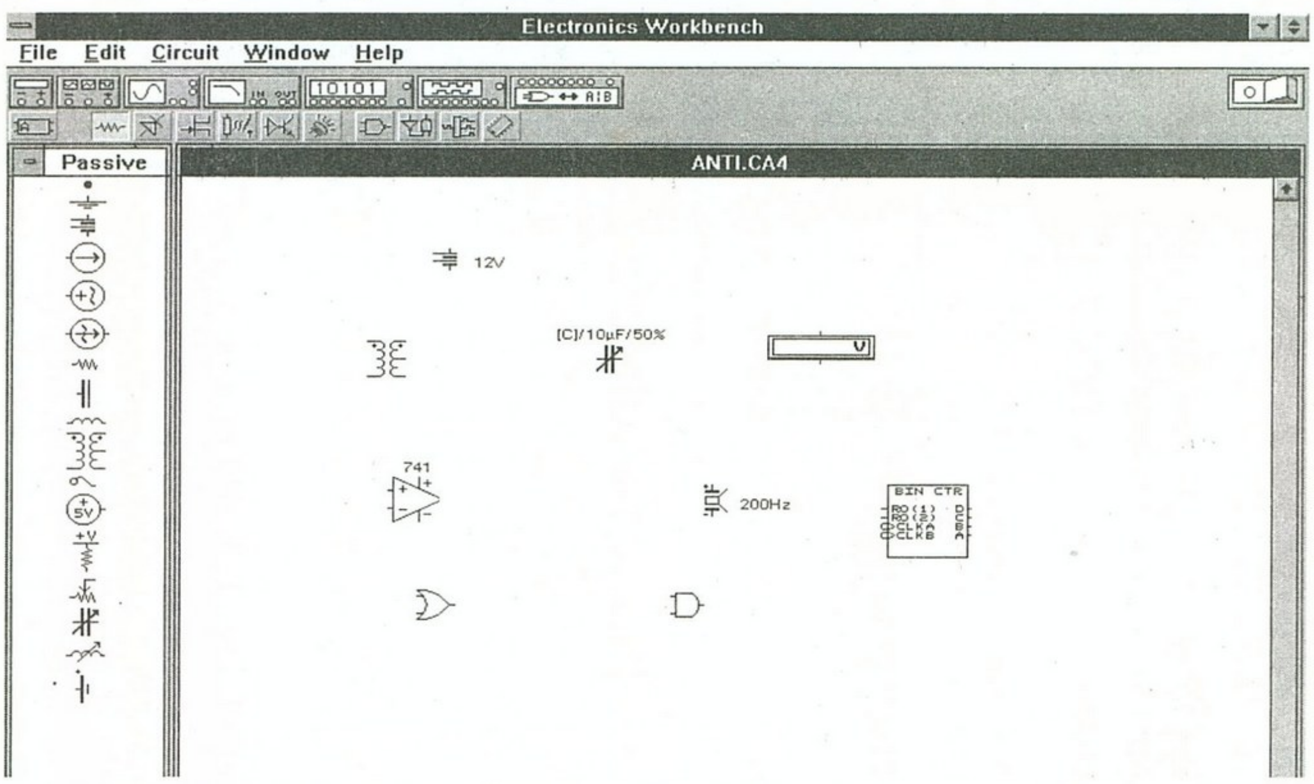
A programhoz segédkönyvek is járulnak a diákok és a tanárok számára, mintakapcsolásokkal, feladatlapokkal együtt.

Bár az analízisszolgáltatások tekintetében elmarad a professzionális szimulációs programoktól (például Tina, Design Center), a valósághoz közelálló látványosság miatt a középfokú szakmai képzés szakmai orientációs időszakában igen jól lehet használni.

A program fejlesztői valószínűleg foglalkoznak a többszörös analízis, a hőfokfüggés, az optimalizálás lehetőségeinek megteremtésével is.

Minimális hardverigény: 386-os vagy 486-os processzor, matematikai társprocesszor, 4 Mbájt RAM, SVGA színes monitor. A program 15 percig teljesértékűen működő demóváltozata és ismertetője megkapható a forgalmazónál.

Kiss Antal



Adatbázis-iskola — II.

Matematikai szabályok és műveletek

A matematika közismert alapműveleteit a számítástechnika sem változtatta meg. Pontosabban a számítástechnika a korábbi szabályokat alkalmazza, de ki lehet emelni néhány olyan dolgot, amely a hétköznapi matematikában kisebb hangsúlyt kap, illetve másképpen működik.

A precedenciaszabály szerint egy egymás után felírt műveletsorból a program kiemeli a magasabb rendűeket a következő sorrendben:

3. Hatványozás/gyökvonás
2. Szorzás/osztás
1. Összeadás/kivonás

Az azonos sorú műveleteket a felírás sorrendjében elvégzi, majd az alacsonyabb rendű műveleteket végzi el a felírás sorrendjében. Tehát először elvégzi az egy sorban talált összes hatványozást, azok eredményeivel az összes szorzást, majd a szorzás eredményeivel a felírt sorrendben összead, illetve kivon. Ha valahol azonos rendű műveleteket talál, ott a felírási sorrend dönt (ez általában nem befolyásolja a végeredményt). Ez a precedenciaszabály, amelyet most egy példával illusztrálunk:

Egy sorban a következő műveletek vannak: 2+5*4–48*12/4+3.

Először elvégzi a szorzásokat, osztásokat: 2+20–144+3. Utána összead, illetve kivon.

A precedenciaszabálytól eltérő műveleti sorrendet zárójelezéssel adhatunk meg. A zárójelek használatában különösebb új nincs, csak két dolgot kell megjegyezni:

— Csakis a normál kerek zárójeleket lehet használni, még akkor is, ha ún. egymásba ágyazott zárójelek szükségeltetnek. Hibás a következő felírás: [12*(a+b)], mivel sarkos zárójelet is alkalmaz, az viszont a különböző programnyelvekben más célt szolgál (például tömbök indexének jelölését). Helyesen ugyanez: (12*(a+b)). A gépírók által (a régi írógépes megszokásból eredően) különösen kedvelt / jel zárójelként nem jöhet szóba, mert ez a jel az osztás jele. A többi zárójel használata a hibajelzés miatt azonnal kiderül, a / esetén viszont nem, csak éppen az eredmény lesz rossz!

— A nyitó és záró zárójeleknek szigorúan párban kell lenniük. A páros megfeleltetést mind a dBase, mind az Excel figyeli, és nem fogadja el a parancsot, ill. képletet addig, míg a páros megfeleltetés nem egyezik. Ez sok és bonyolult, részben egymásba ágyazott szerkezetű képlet esetén szokott igazán gondot okozni. Ilyenkor nagyon figyelmesen kell javítani, mert könnyen bekövetkezhet, hogy a rossz helyre beszúrt zárójel formálisan nem kifogásolható, tehát hibaként már nem fog mutatkozni, a képlet mégis hibás eredményt ad. Például az előző képlet módosításával kapott (12*(a)+b) formula formálisan megfelelő, csak az eredmény lesz rossz.

Műveleti jelek és alkalmazásuk

+, – az összeadás és kivonás,
*, / a szorzás és az osztás,
^ a hatványozás.
=, <, > jelek és ezek kombinációi: <=, >=, < > az egyenlőség, a kisebb-nagyobb relációk beállítása. A dBase a < > jel helyett a #, vagyis a nem egyenlő jelet is elfogadja.

A dBase ezeken kívül a logikai operátorokat, a keresési operátort és a makróoperátort is használja, ezért azok más célra nem használhatók:

.and., **.or.**, **.not.** **!** a logikai operátorok. (Az Excel ezek helyett függvényt alkalmaz.)

$ jel a keresési operátor,

& a makróoperátor.

A magyar ékezetes billentyűzet a ^ karaktert nem tartalmazza, de beírására több lehetőség is van. Legbiztosabb az Alt gomb lenyomva tartása mellett a jobb oldali számbillentyűzeten a kódszám, a 94 begépelése, majd az Alt elengedése. A legelterjedtebb billentyűzetátdefiniálási lehetőség az eredeti amerikai és a magyar klaviatúra között a Ctrl-Alt-F1, és vissza a Ctrl-Alt-F2. (Az amerikai kiosztásban a Shift-6 adja a ^ jelet.) Újabban gyorsan terjed a Multikey billentyűzetdefiniáló is, ott a magyar billentyűzetkiosztásból ki sem kell lépni, csak a Shift-CapsLock-6 kombinációt leütni. A magyar Windows alatt telepíthető a RUSLAT.EXE program, amelynek ikonja egy Pri/Sec feliratú ikon. Az Alt-Tab ismételt megnyomásával rálépünk, ekkor a Sec ikon látszik, és az amerikai billentyűzet él. Visszacsinálni értelemszerűen lehet.

Blokkdiagramok — a programozás alapjai

Elöljáróban le kell szögezni egy általános érvényű igazságot: minden olyan eljárás programozható, amelyet — bármilyen bonyolultan is, de — egyértelműen meg lehet fogalmazni szavakkal és mondatokkal. Ugyanakkor nem lehet programozni azt az eljárást, amelyre az előbbi állítás nem igaz. (Az előbbi kitételt egy elrettentő, de nagyon gyakori példával illusztrálom. Egy munkafolyamatban általában — nevezzük így — „A" eljárás szerint végzik a dolgozók a munkájukat, de néha a „B" eljárást követik. A kérdésre, hogy mikor „A" és mikor „B" a követendő eljárás, a válasz: hát, ha a főnök azt mondja, meg ha úgy érezzük, hogy a „B" a jobb, akkor azt választjuk. Nos, a főnök szeszélye és a beosztott érzése nem programozható. Ha világosan meg lehet fogalmazni, hogy milyen feltételek határozzák meg az „A" vagy „B" választását, az már programozható.

Amikor egy számítógépes környezetben kíváncsiak vagyunk valamire — tetszik, vagy nem tetszik —, programot vagy ahhoz hasonló valamit kell írnunk. Az első bekezdés értelmében az első lépés mindig az legyen, hogy megfogalmazzuk: mit is akarunk látni, mire vagyunk kíváncsiak. A második lépés azután az, hogy megfogalmazzuk: ehhez milyen adatok kellenek, és azokból milyen eljárásokkal tudjuk kihozni az eredményt.

Ez így nagyon bonyolultnak hangzik, de a valóságban sokkal egyszerűbb végrehajtani. Például tudjuk, hogy van egy személyzeti adatbázisunk. Ne legyünk kíváncsiak csak úgy általában, hanem mindig konkrét dolgot kérdezzünk magunktól. Például tudni szeretnénk szervezeti egységenként (főosztály, osztály) az átlagbéreket. Ehhez az adatbázisban benne kell lennie annak, hogy ki melyik szervezetnél dolgozik, és hogy mennyi a bére. Ha valamelyik adat nincs az adatbázisban, meg kell adnunk azt. Ezután már csak összegezni kell a béradatokat szervezeti egységenkénti külön-külön változóba vagy két tömbbe — ahol az egyik tömb a bérek összege, a másik pedig a szervezeti egység létszáma, a tömbök indexe pedig a szervezeti egység sorszáma —, majd az átlagokat kiszámítani a következő műveletekkel: az első tömb x. elemét elosztjuk a második tömb ugyanannyiadik elemével. (Természetesen az összes tömbelemre vonatkoztatva, külön-külön.)

A lépéseket ezután addig finomítjuk, míg az általunk már ismert parancsokkal le nem tudjuk írni az eljárást.

Más eljárás ugyanerre a célra blokkdiagram felvázolása. A blokkdiagram szerkesztésének szabálya, hogy az egyes lépéseket különböző alakú geometriai idomokba írjuk, és azokat a folyamat természetes irányának megfelelően nyilakkal összekötjük. Az egyértelmű, egyszerű lépéseket téglalapba, az összetett (további blokkdiagrammal leírható) lépéseket kettős téglalapba, az eldöntendő (a döntési feltételek kiértékelése után többelágazásos) feladatlépéseket rombusz alakzatba írjuk, ahol a rombusz sarkai jelentik az igaz-hamis ágat. Az eredmény kiíratását egy papírlapot jelképező alakzatba írhatjuk.

A felírást addig finomítjuk, amíg valamennyi lépés programnyelvi parancsokkal mint lépésekkel le nem írható. Példaként álljon itt az adó kiszámítására vonatkozó eljárás, először a „mondatok írásának" variációjában, utána blokkdiagramban.

Adó — a dBase esetére

Első közelítés

— Az éves bevétel bekérése.

— A bevétel összehasonlítása az adótábla soraival.

— A talált érték alapján az adó kiszámítása.

Második közelítés

— Az éves bevétel bekérése.

— Adatbevitel képernyőről.

— A bevétel összehasonlítása az adótábla soraival.

— Vond le a bevételből a tb-járulékot (10%) és a munkanélküli járulékot (0,5%).

— Soronként: a csökkentett bevétel nagyobb-e a sornál? Ha igen, továbblép, ha nem, megadja az alapadót, a kulcsot, majd kilép.

— A talált érték alapján az adó kiszámítása.

— Vond le a csökkentett bevételből a sor alapösszegét.

— A maradékot szorozd meg a sor kulcsával.

— A kapott eredményhez add hozzá a sor alapadóját.

Harmadik közelítés

(Megjegyzés: már most leírok néhány olyan dBase-parancsot — itt ilyen az input —, amelynek részleteit a dBase használatát bemutató szakkönyvből lehet megtudni, de a későbbiekben én is adok kicsit részletesebb magyarázatot. A működő program a lemezmellékleten *ADO.PRG* néven található meg.)

```
input "Kérem az éves
bevételt" to bevetel


adoalap=bevetel–
-bevetel*0,1–bevetel*0,005
```

Az előbbi sor a számítástechnikában járatlanok számára furcsa. A gép úgy működik, hogy először megkeresi és tárolja az egyenlőség jobb oldalán álló változókat (itt: bevetel) a későbbi műveletek számára, azokkal a matematika precedenciaszabálya szerint elvégzi a műveleteket, majd az egyenlőségjel bal oldalán álló változónak adja azt eredményül. Tulajdonképpen érthetőbb lesz a dolog, ha az egyenlőségjel helyett itt a „legyen egyenlő" kijelentést használjuk a különböző esetekre:

```
Ha az adóalap < 110000
        savhatar=0
        alapado=0
        kulcs=0
Ha az adoalap < 150000
        savhatar=110000
        alapado=0
        kulcs=0,2        &20 %
Ha az adoalap < 220000
        savhatar=150000
        alapado=8000
        kulcs=0,25       &25 %
Ha az adoalap < 380000
        savhatar=220000
        alapado=25500
        kulcs=0,35       &35 %
Ha az adoalap < 550000
        savhatar=380000
        alapado=81500
        kulcs=0,4        &40 %
Egyébként:
        savhatar=550000
        alapado=149500
        kulcs=0,44       &44 %
```

ado=(bevetel–savhatar)*kulcs+alapado

(Lásd a blokkdiagramokat.)

Adó — az Excel esetére

Első közelítés

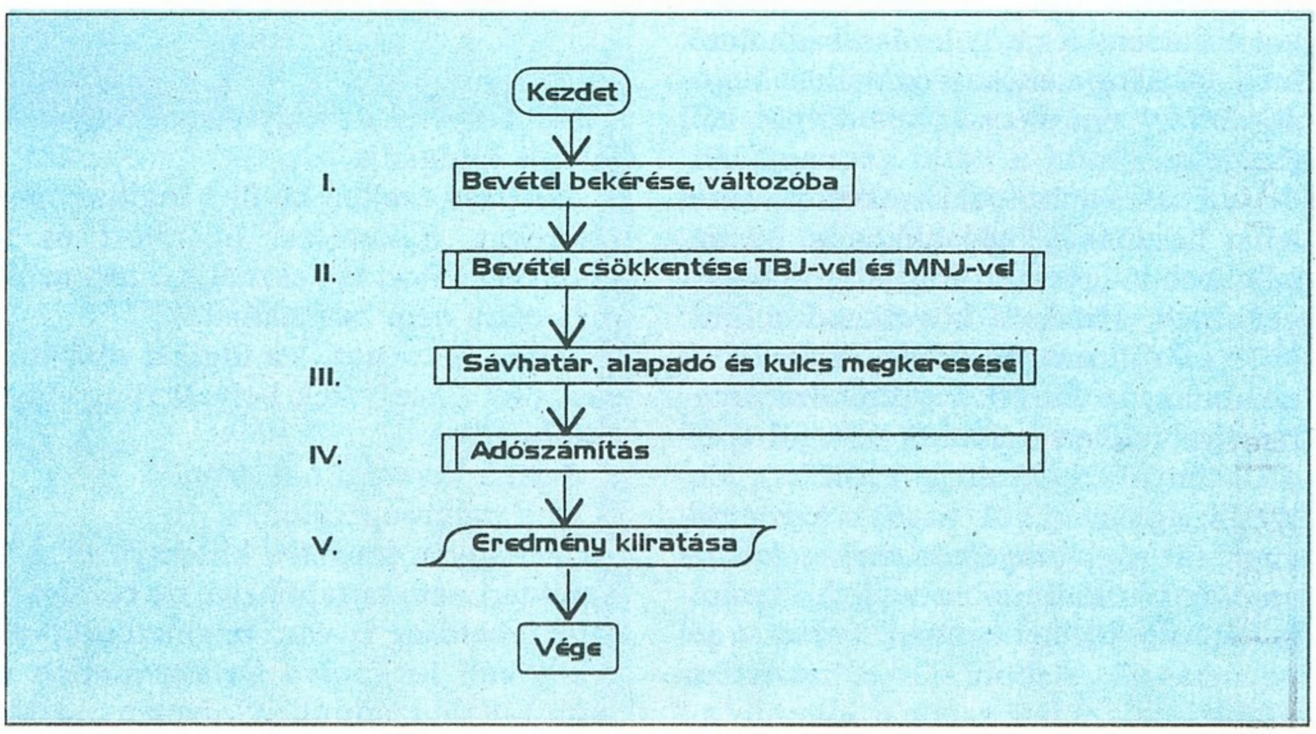
— Az éves bevétel bekérése.

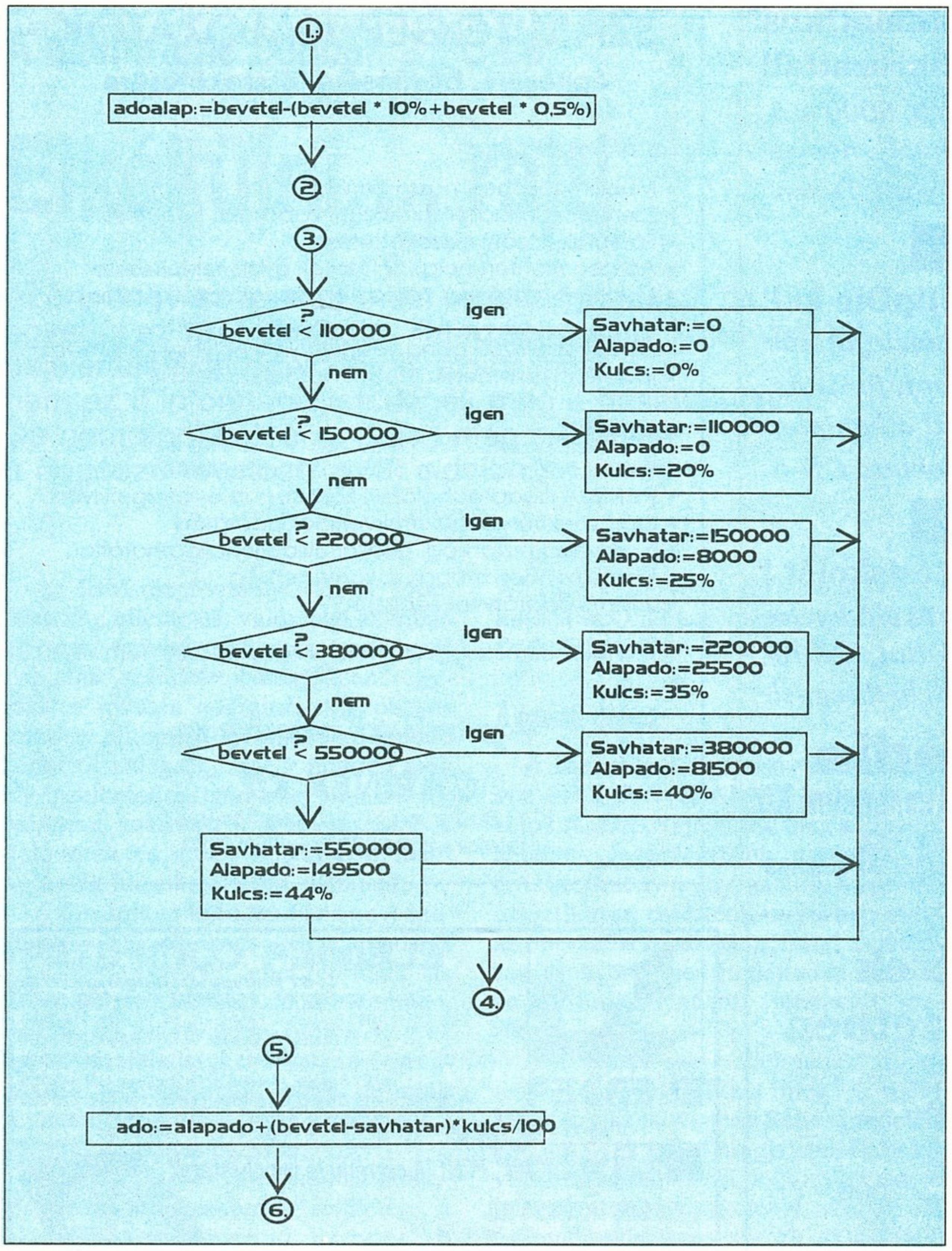
— A bevétel csökkentése tb-járulékkal és munkanélküli járulékkal.

— A bevétel összehasonlítása az adótábla soraival, a talált érték alapján az adó kiszámítása.

Második közelítés

— Az éves bevétel bekérése





— Adatbevitel az egyik cellába.

— Az adóalap kiszámítása: a bevétel csökkentése.

— Vond le a bevételből a tb-járulékot (10%) és a munkanélküli járulékot (0,5%).

— Tedd az Adóalap nevű cellába.

— Az adóalap összehasonlítása az adótábla soraival.

— Soronként: az adóalap nagyobb-e a sorbeli sávhatárnál?

Ha igen, az eredménycella 0, ha nem, megadja az értéket, az Alapadóhoz hozzáadja a sorbeli kulccsal szorzott Sávhatárt az eredménycellában.

Az eredménycellák közüli legnagyobb érték az adócella tartalma.

A tábla első felírása után zavarónak tűnt, hogy a hasznos — a legnagyobb — értékig az összes cella tartalma látható volt. Bár az eredményt ez nem befolyásolja, mégis bevettem egy további feltételt is: az (Adóalap–Sávhatár) kisebb a sávhatárnál. (Tessék kipróbálni enélkül!)

Harmadik közelítés

Szöveg helyett egy Excel-táblázat álljon itt, először mint számolótábla, utána a képleteket tartalmazó tábla.

A nem teljesen látszó sorok:
Bevétel–(Bevétel*Tbj+Bevétel*Mnj)
HA(ÉS(Adóalap>Sávhatár;(Adóalap –Sávhatár)<Sávhatár);Alapadó+(Adó alap– Sávhatár)*Kulcs;0)

A tábla a lemezmellékleten *ADO-SZAM.XLS* néven megtalálható. Kipróbálásához a bevételt a B2 (piros alapon fehér betűk) cellába kell beírni. Az ott látható 567 000 Ft általam véletlenszerűen kitalált szám.

Ferenczi Gábor

Bevétel beírása:		Munkatábla:				Levonás tábla	
Bevétel:	567 000 Ft	Sávhatár:	Alapadó:	Kulcs:	Eredmény:	Társ.b.j.	Munkan.j.
		0	0	0%	0	10,00%	0,50%
Eredménytábla:		110 000	0	20%	0		
Adóalap:	507 465 Ft	150 000	8 000	25%	0		
Adó:	132 486 Ft	220 000	25 500	35%	0		
		380 000	81 500	40%	132 486		
		550 000	149 500	44%	0		

Bevétel beírása:		Munkatábla:				Levonás tábla	
Bevétel:	567000	Sávhatár:	Alapadó:	Kulcs:	Eredmény:	Társ.b.j.	Munkan.j.
		0	0	0	=IF(AND(Ad	0,1	0,005
Eredménytábla:		110000	0	0,2	=IF(AND(Ad		
Adóalap:	=Bevétel-(Bevétel*	150000	8000	0,25	=IF(AND(Ad		
Adó:	=MAX(Eredmény)	220000	25500	0,35	=IF(AND(Ad		
		380000	81500	0,4	=IF(AND(Ad		
		550000	149500	0,44	=IF(AND(Ad		

Nyelvet tanul és ...

... unalmas a szótanulás?

A legnépszerűbb!

Tanuljon a PicDic tematikus képes szótársorozattal!

83 témakör, 200 kép, 5000 szó, élő hang, tesztek az angol-, német- és francia-magyar CD-ken.

... nem érti az élő nyelvet?

Sokoldalú!

Tanulmányozza a ClipDic-kel!

Tematikus videoklipek, visszajátszás mondatonként, szavanként, fordítás, nyelvtan, és hétféle teszt a komplex angol nyelvoktató CD-n.

.. szabadulna hibás kiejtésétől?

A legteljesebb!

Járja ki az Angol Kiejtésiskolát !

270 oldal az angol kiejtésről élő nyelvi példákkal, sok-sok gyakorlattal, mikrofonos kiejtésösszehasonlítási lehetőséggel.

Profi-Média Kft.

Profi-Média

Számítástechnikai és Kereskedelmi Kft.

H-6500 Baja, Kölcsey u. 112.
Tel./Fax.: (36) 79-325 983, Tel.: (36) 30-466 339

Rádiófrekvenciás adatátvitel

VEZETÉK NÉLKÜLI SZÁMÍTÓGÉP-HÁLÓZATOK

építésére, bővítésére, összekötésére

Lehetőséget biztosít:

- Mobil munkahely használatára
- Szerver-munkaállomás közötti kapcsolat létesítésére
- Épületek közötti összeköttetésre
- Ad hoc (konferencia) hálózatok gyors telepítésére
- Meglévő vezetékes hálózatokhoz való csatlakozásra

- Nincs kábelezési gond, megszűnik a helyhez kötöttség
- Ipari környezetben is megbízhatóan működik

Jellemzői:

- Széleskörű hálózat- és protokoll-kompatibilitás
- PCMCIA adapter, hálózati kártya, hub és bridge kivitel
- Szórt spektrumú rádiófrekvenciás adásmód
- Nagy adatbiztonság, gyakorlatilag lehallgathatatlan
- Kiemelkedően magas zavarvédettség
- 2 Mb/s adatátviteli sebesség

A Hírközlési Főfelügyelet által engedélyezett termék.

Magyarországi forgalmazója:

CARINEX Kft.

1113 Budapest XI., Nagyszőlős u. 51.
Telefon: 252-0221 Fax: 251-9280

INFORMÁCIÓKÉRÉS: 0805 ▲

Ügyviteli szoftverek WINDOWS-környezetben

Könyvelési rendszerek

Kettős könyvelés	25 000,– Ft
Naplófőkönyv, pénztárkönyv	10 000,– Ft
Komplett pénzügyi rendszer	25 000,– Ft
Bér- és munkaügyi rendszer	30 000,– Ft
Tárgyi eszközök, elszámolás, nyilvántartás	30 000,– Ft
Számlázás, készletnyilvántartás	30 000,– Ft

Egyedi WINDOWS-felületű rendszerek megvalósítása

CompConto Kft.
1165 Budapest, Jókai u. 4.
Telefon: 252-3093

INFORMÁCIÓKÉRÉS: 0806 ▲

ELENDER COMPUTER

1087 Budapest, Hungária krt. 8.
Tel.: 134-5214, 114-0532 Fax: 133-4347

1092 Budapest, Ferenc krt. 16. Tel./Fax: 218-2858
1134 Budapest, Csángó u. 13. Tel./Fax: 270-3097
4029 Debrecen, Piac u. 57. (Amfóra udvar) Tel./Fax: (52) 413-795
6721 Szeged, Madách u. 15. Tel./Fax: (62) 310-269
8200 Veszprém, Zrínyi u. Botev üzletház Tel./Fax: (88) 428-235

ELENDER

Nyitva: hétfőtől péntekig 9-17 óráig

Maxtor PCMCIA cserélhető winchesterek



131 MB, PCMCIA III.
Operating Shock: 120 Gs
Non-operating Shock: 600 Gs
MTBF: 300.000, 14 ms., 10x53x84 mm

MobileMax 20 MB

Flash card-ok

2 MB 4 MB 8 MB 12 MB 16 MB 20 MB

ENHANCED IDE winchesterek

Maxtor	*7546A: 546 MB, 3.5", 12 ms*	*25.900 Ft.+áfa*
Maxtor	*7850A: 850 MB, 3.5", 11 ms*	*32.900 Ft.+áfa*
Maxtor	*71050A: 1.05 GB, 3.5", 11 ms*	*41.900 Ft.+áfa*
Maxtor	*71260A: 1.26 GB, 3.5", 11 ms*	*43.900 Ft.+áfa*

MAXOPTIX T4 OPTIKAI DRIVE



Maxoptix

Paraméterek:

- ***1.3 GB***
- ***34 ms hozzáférési idő***
- ***2.2 MB/s***
- ***SCSI II.***
- ***1 MB Cache***
- ***41x146x203 mm***

Biztonság:

- ***100.000 óra MTBF***
- ***Novell bevizsgált***

INFORMÁCIÓKÉRÉS: 0815 ▲

A mainframe bebújik a PC-be

Elkékülni..., rajt!

Azért, hogy ez a rovat most rendhagyó külsőt ölt, egyedül az IBM okolható. Ha szépen elaprózták volna bejelentéseiket egész évre, akkor most nem az ő információik töltenék meg e hasábok javát. De nem aprózták el... (Lásd még cikkünket a Fogódzó rovatban a 43. oldalon.)

Az IBM megfogalmazásában: „Két, látszólag ellentétes, valójában azonban egymást erősítő tendencia érvényesül napjaink számítástechnikájában: egyrészt a minden eddiginél nagyobb és minden eddiginél gyorsabban fejlődő funkciógazdagság, amely nemrég még elképzelhetetlen igényeket hivatott kielégíteni, másrészt a szabványosításra való törekvés, amely nélkül az új technológiák kezelhetetlenné válnának.

A dinamikus fejlődés és az egységesítésre való törekvés ellentmondását csak egyféleképpen lehet feloldani: ha komplex problémával állunk szemben, osszuk fel azt részekre, oldjuk meg az egyes részfeladatokat a lehető legmagasabb szinten, majd előre definiált felületen kapcsoljuk össze a részmegoldásokat egységes egésszé.

Ennek a jól bevált elvnek az átültetése a számítástechnika területére a kliens/szerver koncepció. Ez az architektúra lehetővé teszi a bonyolult rendszerek használatának szabványosítását, ugyanakkor meghagyja a fejlesztők szabadságát és megóvja a felhasználókat attól, hogy besétáljanak egy egyedül üdvözítőnek kikiáltott megoldás zsákutcájába."

A fenti elv szellemében az IBM a nagyszámítógépeknél kiforrott technológiákat alkalmazza a PC-fejlesztésben, a sokmillió felhasználó igényei szerint formálódott felhasználói csatolók pedig megjelennek a közép- és nagygépes környezetben. Miközben a régi központi gépek funkcióit lokális hálózatok veszik át, az új mainframe-ek megjelennek a LAN-okon. Az AS/400-ba integrált PC-s hálózati kiszolgálókkal így természetesen férnek meg a PC-be épített mainframe-ek.

Hardver oldalról a PowerPC processzorcsalád a közös alap, de a közös alapok nem jelentenek egyformaságot. Az IBM a szerverek széles választékát kínálja az OS/2 LAN Servertől az AIX-en és OS/400-on át az MVS-ig.

A fekete doboz

A dinamikus fejlesztés következtében évről évre új AS/400 modellsorozatok (C,D,E,F) jelentek meg a piacon, amelyek legjellemzőbb mutatója, az ár/teljesítmény arány, az idők folyamán drasztikusan csökkent, miközben mind az operációs rendszerek, mind az alkalmazások az összes modellen korlátozás és újrafordítás nélkül futtathatók maradtak.

A befektetések védelmének figyelembevételével alkotta meg az IBM 1994-ben az Advanced Series modelleket, amelyek külső megjelenésükön kívül (fekete kisméretű dobozok) számos jelentős többletfunkcióval rendelkeznek, operációs rendszerük az OS/400 V3R1. A mostani júniusi bejelentés a 64 bites RISC PowerPC-alapú, OS/400 V3R6 operációs rendszerű gépekre terjed ki.

Az, hogy 64 bites a hardverarchitektúra, és az operációs rendszeren kívül maguk a meglévő alkalmazások képesek kihasználni az új hardverkörnyezetet, egyáltalán nem csoda, hiszen az AS/400-at eredetileg 128 bites architektúrájúra tervezték! A modellek a PowerPC 620-as processzor AS/400-hoz igazított változatával készülnek.

Az Advanced Series System családja kibővült a 400-as, 500-as, 510-es, valamint az 530-as modellekkel. A szerverek kategóriájában a 40S, 50S, valamint az 53S gazdagítja a kínálatot. Az új modellek nem egyszerűen felváltják a korábbiakat, hanem kiegészítik a meglévő kínálatot. Ezáltal az AS/400 család egy széles tartományban finoman hangolható.

A RISC architektúra megjelenésével egy időben a gépek kiépíthetősége jelentősen változott. A maximális memória értéke meghaladhatja a 4GB-ot, a lemezkapacitás az 500 GB-ot, a kommunikációs vonalak száma a 200-at, a maximális felhasználók száma pedig elérheti a 7000-et! A kiépítettségen kívül a teljesítménynövekedés is szembeszökő: 40%-kal növekedett a hagyományos 48 bites architektúrához képest! Újdonság még az alternatív IPL eszközként is használható CD-ROM.

Az operációs rendszer magját C++-ban fejlesztették, előrevetítve a fejlődés további irányát az objektumorientált rendszerek felé. Többféle operációs rendszer környezetet kínálnak „dobozon belül".

A különböző architektúrák, mint például a Unix, LAN server, Novell Netware 4.1, Lotus Notes stb. egyesítése közös rendszerfelügyelet alatt soha nem látott mértékben valósítja meg ezek integrációját.

Az AS/400 alkalmassá vált a WWW-szerver szolgáltatások ellátására is. A WWW böngésző programok a világ bármely pontjáról az Internet közbeiktatásával válhatnak az AS/400 kliensévé.

OS/2-, valamint Windows-ablakokban valódi grafikus felületen keresztül érhető el az OS/400 valamennyi szolgáltatása.

Chipetnyi mainframe

Nemrég a mainframe még óriási méretű számítógép volt, amely sok helyet foglalt, vízhűtést igényelt, és kizárólag nagyvállalatoknál, államigazgatási és tudományos intézményeknél fordult elő. A System/390 átalakult a tegnap mainframe-jéből a ma vállalati szerverévé. Információt szolgáltat gyorsan és megbízhatóan az év minden napján, 24 órában. 1994-es bejelentése óta több mint ezer System/390 Parallel Enterprise szervert helyeztek üzembe. Akkori elképzelésüknek megfelelően — egy-másfél évenként akarják megújítani a családot — most időszerű új processzorral ellátni a 9672-es sorozatot.

Az egyprocesszoros modell teljesítménye mintegy 1,6-szorosa a tavalyi modelleknek, ilyen CMOS-alapú chipprocesszorokból pedig nemcsak 6, hanem 10 helyezhető egy gépbe, mintegy 2,5-szeresére növelve a gépcsalád legnagyobbikának teljesítményét. Ez a technológia jelentősen csökkentette a mainframe gépek költségeit, a skálázhatóság lehetőségei is kiszélesedtek, hiszen a jelenlegi sorozat 12 modellt

tartalmaz. A tavalyi 9672-es központi egységek egyszerűen átalakíthatóak a jelenlegi modellekre.

Ezek a gépek már Magyarországon is megjelentek, kiváltva ezzel a régi 43xx gépcsalád tagjait. Szintén 1994-ben jelentették be a RAMAC lemezcsaládot, amely hibatűrő RAID-5 technológiát használ. Akkor 3,5 inches, 2 GB-os lemezekből építették fel a rendszert — egy embernagyságú rackbe 90 GB-ot lehetett tenni. Most a RAMAC is megújult, hiszen mostantól rendelhető az Ultrastar XP 4 GB-os lemezekkel, így 180 GB-ra növelve az egy rack-ben elhelyezhető kapacitást. A rendszer új, gyorsabb processzort használ a RAID-5 technológia megvalósításához, többszintű, max. 2 GB-ig növelhető író/olvasó cache-t tartalmaz, a technológia pedig a 24 órás rendelkezésreállást valósítja meg. A beruházásvédelmi szempont értelmében a tavalyi 2 GB-os lemezeket tartalmazó fiókok keverhetőek az új fiókokkal.

Vegyes házasság

Igazi újdonság a PC Server 500 System/390. Ez a rendszer, amely egy Pentiumos PC és egy nagygépes processzor házassága, lehetővé teszi a nagygépes fejlesztők számára az olcsó, gyors fejlesztési lehetőséget.

A PC Server 500 System/390 meglévő nagygépes felhasználók számára külön fejlesztőgép installálását teszi lehetővé; a régi 4341, 4361 és 9373/75-ös géppel rendelkező felhasználók számára költségkímélő, modern eszközre való áttérést; ezenkívül olyan kliens/szerver, illetve osztott feldolgozás megvalósítását, ahol a kliensek is mainframe kategóriájú gépek. A gép futtatja a jelenlegi MVS/ESA, VM/ESA és VSE/ESA operációs rendszereket, saját nagygépes processzora és memóriája van, a hibatűrő (RAID-1, RAID-5) lemezalrendszert (max. 38 GB) a PC szolgáltatja.

Ugyanakkor a Pentium processzor és a hozzátartozó memória OS/2 operációs rendszert futtat, alkalmas LAN szerver (Novell, OS/2 stb.) funkciók ellátására, és kiszolgálja a mainframe rész I/O igényét. A gép tartalmaz beépített CD-ROM-meghajtót, a felhasználó a dokumentációt online tanulmányozhatja. A gép ideális eszköze a nagygépes alkalmazásfejlesztésnek, kombinálva a Visual termékcsalád objektumorientált lehetőségeivel.

A kapcsolatot a Communication Manager/2 biztosítja — csakúgy, mint valós környezetben a nagygép és a LAN között. A nyílt rendszerek megvalósítása a mainframe-ek operációs rendszerében is tükröződik. Az Openedition kiegészítés az MVS/ESA és VM/ESA operációs rendszerekhez magában foglalja a SPEC1170 és az XPG4 által előírt követelményeket, megvalósítva ezzel több mint 1100 Unix-funkciót.

Ezzel lehetővé válik számos Unix-alkalmazás átvitele a nagygépekre, kiegészítve ezeket az alkalmazásokat a mainframe nyújtotta megbízhatósággal, biztonsággal és teljesítménnyel. Természetesen az alkalmazások használhatják párhuzamosan a Unix és a hagyományos MVS, illetve VM lehetőségeket, kombinálva azok összes előnyeit.

Az IBM idén tervezi az MVS/ESA rendszer hivatalos Unix rendszerré nyilvánítását (Unix branding); e minősítéssel jelenleg a világon még igen kevés Unix rendszer rendelkezik. Ezek a bejelentések mind azt mutatják, hogy a nagygépes platform él és növekszik. Egyre több felhasználó ismeri fel a vállalati szerver fontosságát, legyen az adatbázis-, fájl-, hálózati, menedzsment- vagy bármilyen egyéb kiszolgáló.

Árcsökkentés a HP-nél

Ha csupán a termékskála egy szűk részét is érintő, mégis jelentős árcsökkentést határozott el a HP. A szürkeskálás HP ScanJet 3P lapolvasó kb. 23%-kal olcsóbban, 68 000 forint + áfás áron kerülhet a felhasználókhoz. A ScanJet 3p fekvőágyas lapolvasó, felbontása 300 dpi-től 1200 dpi-ig bővíthető, befogadási mérete max. A/4-es. A 20 oldalas automatikus lapadagoló opció áfa nélkül 29 500 forintba kerül.

A beépített lapolvasó szoftverek — PictureScan, PicturePlace, PicturePlace Editor — felépítése olyan, hogy egyetlen egérkattintással lehessen vezérelni a beolvasást. Van még hozzá egy másoló segédprogram, amelynek segítségével ugyancsak egyetlen egérkattintással az irodai nyomtatóra küldhető a beolvasott kép. A HP ScanJet 3p-t PC-hez Recognita Select 2.0-val, Macintosh-hoz WordScan 3.0 OCR szofverrel szállítják.

Világcsúcs a telefóniában

Ha már a magyar telefónia kiépítettsége, fejlettsége és szolgáltatási színvonala nem ostromolhat — az optimisták szerint egyelőre... — csúcsokat, legalább egy üzletkötés nagyságrendjéről elmondható: „hát ilyen még nem volt".

A Motorola ugyanis 1 milliárd dollár értékben építhet ki a Matáv részére rádiós telefonkapcsolatot a meglévő központokhoz (900 MHz-es TEX analóg technológia). A vezeték nélküli kapcsolat két év alatt 200 000 új előfizető bekapcsolását teszi lehetővé azokon a területeken, ahol a vezetékfektetéses megoldás időben és pénzben jóval nagyobb ráfordításokkal járna.

A Motorola magyar beszállítókkal is együttműködik, a Hungarocom például hálózati komponensekkel látja majd el a projekt kiszolgálására alakuló magyar Motorola-leányvállalatot, amely a tervek szerint 200 hazai szakember részére kínál majd munkalehetőséget.

Nemzetközi „hajtóvadászat"

A Novell európai szoftverjogi csoportjának több hónapos nyomozómunkája eredményeképpen sikerült felderíteni egy olyan csatornát, amelyen keresztül hamisított Novell-termékek jelentek meg a közép-európai régióban. A forrás Lengyelország, illetve egy hollandiai raktárbázis volt, de a termékek eljutottak Csehországba, Bulgáriába és vélhetően Magyarországra is.

A hamisítást azért is volt nehéz kiszűrni, mert nem szoftvermásolás, hanem OEM-„átöltöztetés" volt a technika. Miközben a Novell bírósági eljárásokat kezdeményez a hamisítók ellen, nem akarja büntetni a félrevezetett felhasználókat: azok számára, akik zöld zárjegy nélkül vásároltak NetWare termékeket az idén, kedvezményes programot hirdet a cég, e program keretében megszerezhetik a hiányzó szoftverlicenceket. Mind az érintettek, mind pedig a termékeik eredetét illetően bizonytalanok számára a Novell budapesti irodája nyújt felvilágosítást (telefon: 266-7770).

CAD/CAM és multimédia

Az őszi számítástechnikai rendezvényeknek mintegy nyitányaként szeptember 12-től 14-ig a Gellért szállóban kerül sorra a CAMP '95 Nemzetközi Szakkiállítás és Konferencia. Az ötödik alkalommal lezajló szakmai találkozó látgatottságát — az 1994-es hibákból okulva — a rendezők és a kiállítók közös levélreklám-akcióval igyekeznek javítani. A helyszín megválasztása megkönnyíti a parkolást a gépkocsival érkezőknek, mert a közelben, a Budapesti Műszaki Egyetem előtti parkolóban lesz elegendő hely. A CAMP szervezője a Scope Rendezvényszolgálati Kft, kivitelezője a Compexpo Kft, fő szponzora pedig a World Computer Graphics Association.

APRÓHIRDETÉSI MEGRENDELŐLAP

Kérem, hogy az Új Alaplap következő számának Mikrobazár rovatában közöljék az alábbi szövegű apróhirdetést:

....

....

....

....

....

(Maximális terjedelem: 300 betűhely)

Alaplap Posta

MEGRENDELÉS

Az Új Alaplap 1995/8. számában ismertetett termékek közül **postai utánvétellel** megrendelem az alább felsoroltakat. Azok árát a küldemény átvételekor a kézbesítési díjjal együtt kifizetem.

1. A 61. oldalon szereplő Automex-válogatásból az alábbi sorszámú **CD-ROM-okat:**

2. A 19-20. oldalon ismertetett **szoftverek** közül az alábbiakat (a túloldalon is folytatható):

Név: ....

(Cég:) ....

Cím: ....

Helység: ....

Irányítószám: ....

Dátum:

/aláírás/

Előfizetés az Új Alaplapra

Az 1995/......... számtól kezdődően előfizetem az Új Alaplap című, havi számítástechnikai folyóiratot ...... példányban, ❏ 1 évre, ❏ 1/2 évre.

Az éves előfizetési díj 3564,– forint.

Az előfizetési díj kiegyenlítéséhez:

❏ **Számlát** kérek (banki átutalással fizetek).

❏ **Átutalási postautalványt** kérek.

Név: ....

(Cég:) ....

Cím: ....

Helység: ....

Irányítószám: ....

Dátum: ....

/aláírás/

INFORMÁCIÓKÉRÉS

Kérem, hogy az itt általam **BEKARIKÁZOTT KÓDSZÁMÚ** hirdetésekkel kapcsolatban küldjenek részemre bővebb tájékoztatást.

0801	0813	0825
0802	0814	0826
0803	0815	0827
0804	0816	0828
0805	0817	0829
0806	0818	0830
0807	0819	0831
0808	0820	0832
0809	0821	0833
0810	0822	0834
0811	0823	
0812	0824	

Beküldhető: 1995. szeptember 30-ig

ÚJ ALAPLAP 1995/8 AUGUSZTUS

FELADÓ Feladáskor kérjük bérmenteséteni!

A)Egyéni érdeklődő:

Név: ....

Cím: ....

Helység: ....

Irányítószám: ....

B) Vállalati érdeklődő:

Cég: ....

Ügyintéző: ....

Cím: ....

Helység: ....

Irányítószám: ....

Telefon/Fax: ....

Bélyeg helye

Új Alaplap szerkesztősége
Pf. 571
Budapest
1538

Belföldön díjmentesen feladható

Új Alaplap szerkesztősége
Pf. 571
Budapest
1538

Minden PC–hez kell egy jó alaplap!

És egy Új Alaplap!

Új Alaplap szerkesztős
Pf. 571

ADÓ **Feladáskor kérjük bérmentesíteni!**

Név: ....

Cím: ....

elység: ....

Irányítószám: ....

Telefon: ....

❑ A hirdetés egyéni és egyedi jellegű, ezért kérem ingyenes megjelentetését. Kijelentem, hogy annak tartalma nem sérti senki szerzői jogát.

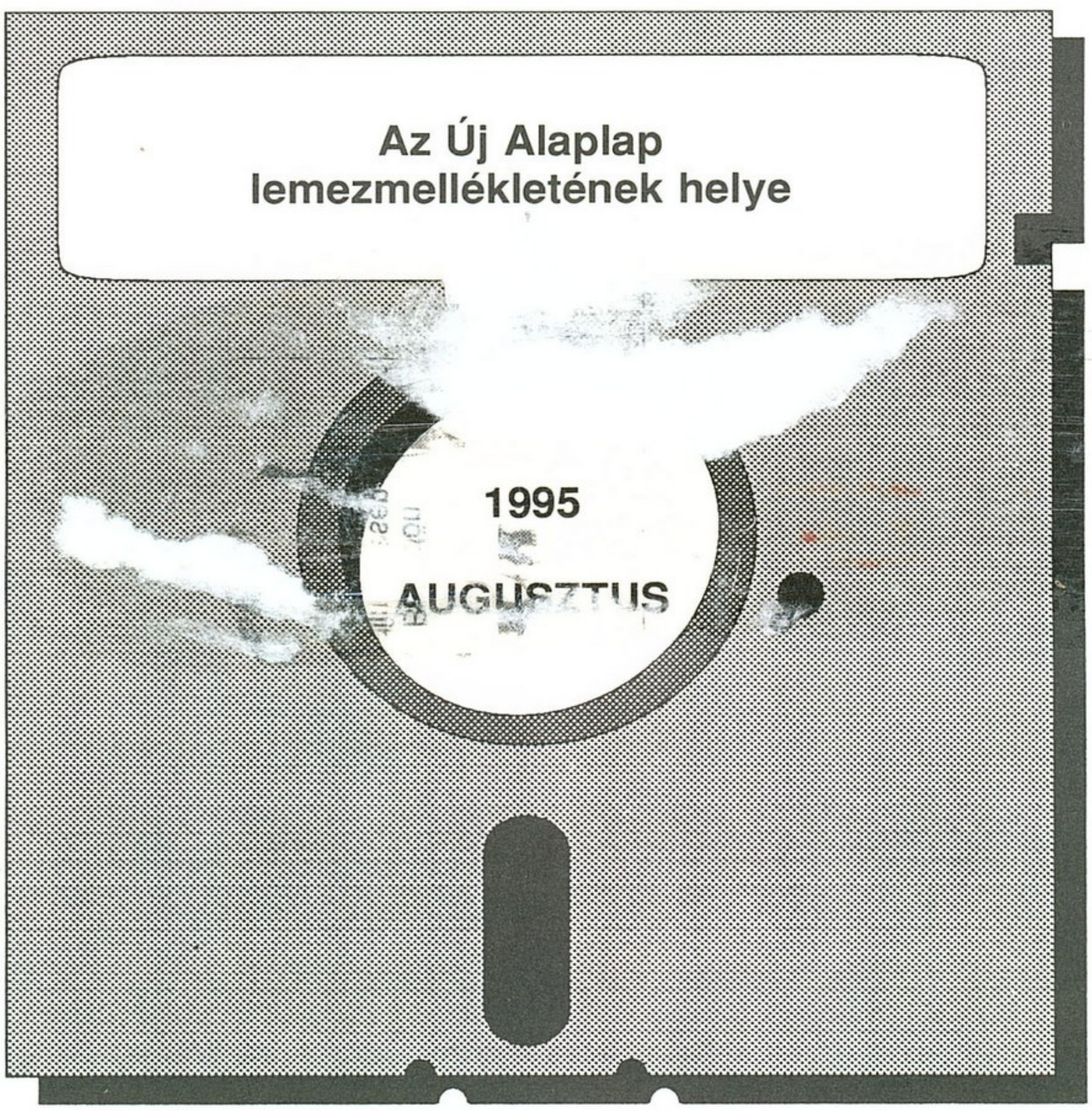
❑ A hirdetés kereskedelmi célt szolgál. Mellékelem a soronként (60 karakterenként) 300 forintnak megfelelő összeg átutalásáról az igazoló szelvény másolatát. címzett: Új Alaplap, 1538 Budapest, Pf. 571, illetve utalásnál: Agrobank 219-93789/2249-6368

Bélyeg helye

Új Alaplap szerkesztősége
Pf. 571
Budapest
1538

Az Új Alaplap lemezmellékletének tartalma 1995 / AUGUSZTUS

- Floppymeghajtók átírása — A_BKONV.EXE, B_AREGN.EXE, BALAP.BAT (Simay Endre István) BATKONV.TXT, SWAPAB.COM [Cikk: 57. o.]
- A hónap témájához: alternatív Program Manager, sőt több... — WAYF#.EXE
- Az adatbázis-iskola illusztrációi — BESZ#.EXE (Ferenczi Gábor) [Cikk: 27. o.]
- Példaprogramok a Fortran-sorozathoz — FOR2#.EXE (Szondi Egon János) [Cikk: 53. o.]
- Prolog nyelvjárások és példaprogramok — MIK1#.EXE, MIK2#.EXE, PPID#.EXE [Cikk: 45. o.]
- A Crush szuperkompresszor — CRUSH.EXE [Cikk: 34. o.]
- Magyar fejlesztésű szövegszerkesztő — BIHA#.EXE (Bihari Zsolt)
- Directory-kezelő program — GDIR.DOC, GDIR.EXE (Andreas-Josef von Gencsy)
- Dámajáték kínai változatban — CHIN#.EXE



NetWare kliens Windows NT-hez

A június végén New Yorkban megrendezett PC Expo alkalmából a Novell bejelentette NetWare Client for Windows NT kliens programcsomagját. Ennek révén a Windows NT alapú munkaállomások felhasználói a NetWare4-gyel való teljes integráció előnyeit is élvezhetik. Hozzáférhetnek annak valamennyi fejlett szolgáltatásához, beleértve az NDS-t (NetWare Directory Services) is. A Novell új, a Windows NT-hez kapcsolódó szoftvere optimalizálja az „NT-munkaállomáson" végzett munkát a NetWare4-gyel egybeépített hálózati szolgáltatásokkal, ugyanakkor a szabványnak tekintett vezető osztott hálózati operációs rendszer használatát teszi lehetővé a Windows NT alkalmazói számára. Az új Novell programot használva az MS-Windows NT 3.5 és NT 3.51 alkalmazói végigbogarászhatják a NetWare címtárát és állományrendszerét, megoszthatják az állományokat és a NetWare erőforrásokat a DOS, MS Windows 3x, OS/2, Macintosh és UNIX felhasználók között; nyomtatást kezdeményezhetnek NetWare alapú hálózati perifériákon; elérhetik a NetWare-alapú alkalmazói programokat és szolgáltatásokat a saját natív NT felületen keresztül. Az NDS integrációja révén az NT-vel dolgozók központi helyről egyszerű módon elvégezhetik a hálózati rendszeradminisztrációt.

Novell bejelentések

A Novell, a Hewlett-Packard és a Xerox közösen jelentette be a NetWare Distributed Print Services (NDPS) újgenerációs nyomtatási szoftverarchitektúrát a NetWare platformra. Az NDPS első kiadása a 4.1-es környezetbe integrált, ahhoz kapcsolódó modulként való megjelenése 1996. első negyedévében várható. A Novell összes programterméke támogatni fogja azt a rövidesen megjelenő OpenDoc szoftverfejlesztő architektúrát, amellyel csökkenthető lesz a szoftverkomplexitás és lehetővé válik, hogy a fejlesztők osztott, keresztplatformos programkomponenseket állítsanak elő.

Az OpenDoc for Windows Developers Release One teljes 32 bites OpenDoc „előverziója" már piacon van. Az OpenDoc for Windows 1.0 fejlesztői kit (SDK) ez év őszén jelenik meg. A cég groupware termékébe, a GroupWiseba beszédfelismerési technológiát épít be. A „hangposta" képességekkel ellátott csoportmunkát támogató Novell termék 1995 őszén jelenik meg a piacon. Kapható viszont már a NetWare for SAA 2.0, a NetWare—IBM központi (host) számítógép közötti kapcsolatot biztosító programtermék új verziója. Ez támogatja az IBM AS/400 gép OS/400 operációs rendszerét, annak V.3. Release 1-ét; a Client Access/400 alkalmazásokat; valamint a korábbi OS/400 V.2-t és a PC Support/400 kliens szoftvereket. Szintén piacon van már a Novell InForms elektronikus dokumentum-(formátum-)kezelő és adatbáziselérési programcsomagjának új, 4.1 változata. Ez a táblázatkezelőkéhez hasonló 58 új formulát tartalmaz.

Városi hálózat CiscoWork 2.0-val

A kaposvári városi hálózatot június végén adta át rendeltetésének a tendernyertes Optotrans Kft. Az alap egy 8,7 km 20 szálas mono illetve kevert mono és multimódusú, valamint 1,2 km 8 szálas multimódusú optikai kábel, mely nagy sávszélességű adatátvitelt tesz lehetővé. A hálózati forgalom irányítását Cisco multiprotokoll routerek (két AGS+, egy C-4000) végzik. A város egyes intézményeiben nagy számú, főleg Novell/NetWare alapú szerver található. Ezen gépek szabályozott keretek között történő eléréséről feltétlenül gondoskodni kellett. Komoly biztonsági követelményeknek is meg kell felelni. Az Internet elérése a HBONE-on keresztül történik, a PATE számítóközpontja és az MTA-SzTAKI közötti 64 kbps-es digitális bérelt vonalon keresztül. A kaposvári hálózaton található összes végpontnak regisztrált IP címe van, így bármelyikről elérhetők az Internet szolgáltatások. A routerek egymás között a Cisco által főleg „intradomain" feladatokra kifejlesztett IGRP routing protokollt használják, míg a SZTAKI és a PATE-n található AGS+ között statikus „útválasztás" van.

A kaposvári városi hálózatot az a CiscoWorks 2.0 felügyelőszoftver vigyázza, amely egy SunNet Manager 2.0-n futó alkalmazás. Segítségével a rendszergazda központi helyről áttekintheti az egész hálózatot, hiba esetén riasztást kaphat és annak helyét pontosan megállapíthatja, távolról konfigurálhatja bármelyik intelligens eszközt, letölthet szoftvert, forgalmi statisztikákat készíthet. A szoftver által szolgáltatott adatok segítségével a hálózat optimális költséggel működtethető és a rendszergazda felismerheti a szűk keresztmetszeteket, illetve iránymutatást kaphat a szükséges bővítési irányokról. A CiscoWorks 2.0 részét képezi a Sybase RDBMS, így a hálózat összes adata, statisztikája más rendszerekbe is átvihető. A hálózat könnyűszerrel bővíthető. A routerek moduláris felépítésűek, így igen egyszerűen „megizmosíthatók", upgradeelhetők. A hálózat és a benne rejlő tartalék lehetővé teszi, hogy egyszerűen legyen átállítható Ethernetről FDDI protokollra. Ez lehetőséget nyitna olyan sávszélesség-igényes kommunikációra is, mint például számítógépes tomográfiai felvételek továbbítása.

IBM—Lotus házasság

Júniusban minden idők legnagyobb értékű (3,5 milliárd dolláros) szoftvercég-felvásárlásra került sor, az IBM tulajdona lett a Lotus cég. Ezáltal annak groupware (Notes) és hálózati (cc:Mail, SmartSuite) és más programtermékei az IBM PC-k hálózati szoftverpiaci pozícióit fogják erősíteni. Egyes amerikai felhasználók első reakciói szerint a „házasság" révén a cc:Mail elektronikus levelező rendszer és a Notes programcsomag a korábbinál jobb terméktámogatása várható. Mások szerint az IBM által birtokolt Lotus cég elveszítheti innovativitását és a piaci változásokra való gyors reagálását. Megint mások azt várják, hogy az IBM vállalati méretű rendszereivel szerzett tapasztalatokat és hálózatmenedzsment támogatást építi be a Notes-ba, ami természetesen a Lotus jelenlegi „zászlóshajó" szoftverének csak a javára szolgálna.

Egyesek arra spekulálnak, hogy a Notes, amely már most is fut OS/2 alapú PC-ken és szervergépeken, natív módon más IBM platformokon (AS/400, RS/6000) is meg fog jelenni. Mindenesetre egy „Microsoft-barát" amerikai cég, a Brainstorm Technology máris megjelent a piacon olyan programcsomaggal, amely összeköttetést biztosít a Notes és a Microsoft Office irodai desktop rendszerek között. Az OfficeLink for Notes szoftver lehetővé teszi az Office használóinak, hogy a Notes adatbázisain tárolt bizonyos dokumentumokhoz anélkül hozzáférjenek, hogy ki kellene lépniük.

Szimbiózisban a Crush

A tömörítők élősködője

A múlt hónapban a PocketD kapcsán megismert Jeff Rollanson írt egy figyelemre méltó tömörítőprogramot is, amely a Crush nevet kapta. Ennek a programnak az alapötlete igencsak régi, és sokaknak ismerős lehet a Unix világából: sokkal jobb tömörítés érhető el, ha egy nagy fájlt tömörítünk, nem pedig sok aprót.

A „sok kicsi helyett egy nagyot" fogással a Crush program még a PKZIP legújabb változatánál is kisebb állományokat készít, és a tömörítésnél talán ez a legfontosabb. Mivel azonban az állományt ilyen módon hoztuk létre, a más tömörítőknél megszokott update műveletekről (törlés, frissítés, stb.) le kell mondanunk.

Bármilyen hihetetlen, de a Crush program használatához szükség van egy valódi tömörítőprogramra is, amely alapbeállításban a PKZIP, de használhatjuk például az ARJ, az LHA, a ZOO közül bármelyiket.

Aki gyakran használja a PKZIP vagy az ARJ programot, az azonnal tudja használni ezt is, míg annak, aki más programokhoz szokott, egy kis tanulásra lesz szüksége. Az utasítás szintaxisa a megszokott: *crush [opciók] archívumnév [fájlnevek] [@lista] [fájlnév]*. Az opciókkal beállíthatjuk, hogy melyik tömörítőt használjuk; megadhatjuk, hogy csak tömörítsen, vagy törölje is a fájlokat; hány különböző tömörítést próbáljon meg, amelyből majd a legjobbat fogja kiválasztani; megadhatjuk, hogy csak akkor tömörítsen a Crush-sal, ha az eredeti tömörítőnél legalább egy előre megadott százalékkal jobb eredményt érünk el; és természetesen egész könyvtárrendszereket is becsomagolhatunk.

Ugyancsak opcióval állíthatjuk be, hogy milyen dátumú fájlokat akarunk tömöríteni. A dátumokat megadhatjuk konkrétan: például *94/4/21*, vagy akár *21feb92*, de természetesen használhatjuk a *2/13/93* jelölést is; relatívan: *-3* három nappal ezelőtt, *m-1* múlt hónapban, *y-2* két éve; sőt megadhatunk egy fájlnevet is, ekkor ennek a fájlnak a dátuma számít. Ha két dátumot adunk meg, akkor e két időpont között készült fájlokat fogjuk tömöríteni. Lássunk egy példát: *crush -:3 backup*, az utóbbi három nap termését tömöríti. Ha felkiáltójelet is használunk, akkor kizárjuk a megadott időpontot, így *crush -!:-0* a nem mai fájlokat fogja tömöríteni. Megadhatjuk, mely fájlokat ne tömörítse, így például a .bak fájlokat a *crush backup -x*.bak* paranccsal kihagyhatjuk az archívumból. Lehetőség van arra is, hogy egy fájlba beírjuk, mely típusú fájlokat akarjuk kihagyni, és ezt adjuk át a programnak.

A Crush-sal tömörített állományok átnézése hosszadalmas feladat, ezért készíthetünk hozzá egy indexállományt, amelyet a d.exe újabb változatai már felhasználnak.

Az állománynév lehet tetszőleges DOS fájlnév, de ha nem adunk meg kiterjesztést, akkor a tömörítőnkre jellemző kiterjesztést kapja az állománynév. Ha név helyett *[ ]*-et adunk meg, akkor a program a rendszeridőből generál nevet, ami mindig más és más lesz, hacsak nem csomagolunk be még valamit két másodpercen belül.

A becsomagolandó fájlok neveinél természetesen használhatjuk a dzsóker karaktert, a csillagot is, ezt viszont nem csak a megszokott DOS-os jelentésével, mert itt a szó kezdődhet csillaggal is: **tutor.**, aminek megfelel például a *ctutor.pas*. (DOS-ban a **tutor.** ekvivalens a *.*-gal.)

Ha nem adjuk meg külön opcióval, a program mindig tárolja a fájlokkal együtt a relatív elérési utat (azaz ha ugyanabban az alkönyvtárban vagyunk, akkor semmit; míg ha két szinttel kintebb, akkor ennek a két alkönyvtárnak a nevét).

A kicsomagoláshoz az Uncrush programot kell használnunk, melynek a szintaxisa: *uncrush [opciók] archívum [célkönyvtár] [fájlmaszk]*. Az archívum ugyanaz a név, amelyet tömörítésnél megadtunk, a célkönyvtár az a könyvtár, ahová kibontjuk az állományt. A tömörített állományban lévő fájlhoz tartozó path hozzáadódik a célkönyvtárhoz, például a *crush backup \wp*.doc* után végrehajtott *uncrush backup mydoc\home* hatására a *mydoc\home\wp* könyvtárba csomagolja ki a DOC fájlokat. A fájlmaszkban szereplő csillagnak itt is megvan a korábban említett tulajdonsága.

Mind a Crush-nál, mind a Uncrush-nál a gyakran használt opciókat egy környezeti változóba tehetjük.

Lássunk egy példát a Crush használatára. Tartalmazza a back.bat a következőket:

```
CRUSH -r:.LASTBAC [] *.DOC *.C *.H
IF ERRORLEVEL 1 ECHO Error !!
IF ERRORLEVEL 1 GOTO Exit
REM>LASTBAC
:Exit
```

Korábban már szerepelt, hogy a [ ] helyett a program egy olyan nevet generál, amely nem eshet egybe a program korábbi futásainak eredményeivel. Ezután veszi mindazokat a .doc, .c és .h fájlokat az egész alkönyvtárrendszerből, melyek frissebbek a lastbac fájlnál. Ha nem történt semmi hiba, akkor létrehozunk egy üres, lastbac nevű fájlt (természetesen az aktuális idővel). Ezért ha ezt a batchfájlt újra futtatjuk, akkor csak a múltkori mentés óta keletkezett vagy módosított fájlokat fogja lementeni.

A batchfájlt egy kicsit módosítva többszintű mentést is megvalósíthatunk.

Ha megtetszett a program, vagy ha nincs elég hely a merevlemezen, a .ZIP fájlokat átkonvertálhatjuk Crush állományokká. Erre érdemes a D.EXE-t használni a következőképpen:

```
MD TMP
CD TMP
D .ZIP/W[PKUNZIP -d $w//CRUSH -ert
$w//PAUSE//D /zd]Rq
```

Minden ZIP állományra a következőket hajtjuk végre: kibontjuk a TMP alkönyvtárba, tömörítjük a Crush-sal, de csak akkor, ha legalább 5 százalékkal kisebb fájlt kapunk, a PAUSE lehetővé teszi, hogy megnézzük, mi is történt, majd ezután megszabadulunk az átmeneti fájloktól.

A bemutatott példákból láthatjuk, hogy a haszonért meg kell tanulnunk egyet és mást, de utána rutinszerűen dolgozhatunk.

Aszalós László—Bakó Mária

CD-ROM-termék születik... — II.

Rögzítés és feldolgozás

A CD-re szánt anyagok sokféleségének, illetve a széles igényskálának megfelelően különböző teljesítményű és minőségű eszközök állnak rendelkezésre az adatrögzítéshez. Először az adatrögzítés célszerű technikáiról szólunk, majd a feldolgozási feladatokra térünk át. Az adatfeldolgozás fázisa magában foglalja mindazon lépéseket, amelyek az adatok javítását, szűrését, adatbázisba szervezésének és megjelenítésének előkészítését jelentik.

A nem túl nagy felbontású és színmélységű képek beolvasására nagy mennyiség esetén a gyors (20-120 A/4-es lap/perc) sebességű lapolvasók ajánlhatók. A minőségi paraméterek javulásával a sebesség csökken, a jó minőséget nyújtó asztali színes lapolvasók a maguk 3-6 lapos percenkénti teljesítményével átlagosnak tekinthetők, a feladatok többségére megfelelnek, s ár/teljesítmény viszonyuk is kiváló. Nagy méret, felbontás vagy színmélység esetén csak a professzionális berendezésektől várhatunk megfelelő eredményt.

Meglehetősen gyakori, hogy a forrásanyag filmen (diapozitívon vagy negatívon) van. Ilyen típusú anyag beolvasásához a lapolvasót fel kell szerelni diafeltéttel, mivel a filmek digitalizálása csak átvilágítással oldható meg. A legtöbb lapolvasó (kivéve a nagysebességűeket) kiegészítő eszközként bír diafeltéttel.

Számítógépi módon

A mozgóképek és a hang digitalizálására speciális számítógépes kártyák, professzionális környezetben célszámítógépek szolgálnak.

A mozgóképek feldolgozására szolgáló hardverek között a célnak a valós idejű feldolgozásra alkalmasak felelnek meg. Fontos paraméter a maximális felbontás, és az eszköz adattömörítési képessége. (Ezek az eszközök állóképek digitalizálására többnyire nem használatosak, mert a lapolvasó technológia — amennyiben a forrás erre lehetőséget ad — jobb eredményt nyújt.)

A hang rögzítése szintén erre a célra szolgáló kártyával történik, vagy digitális vágóasztalról érkezik. A képek, a hang rögzítése igen nagy odafigyelést, fajlagosan nagy élőmunka-ráfordítást igényel.

A források minősége szinte mindig szükségessé teszi a kézi hangolást, beállítást. Nagy adatmennyiségek esetén ez lehetetlen, ilyenkor az automatikus minőséghangolás alkalmazható, de ez minőségi kompromisszumokkal jár. Az adatbevitel idő- és költségkeretének megtervezésénél erre tekintettel kell lenni.

A felsorolt multimédia-adattípusok nemcsak digitalizálás, hanem számítógépes generálás útján is előállíthatók. A számítógépes grafika, animáció, a digitális hanggenerátorok szintén szolgáltathatnak kiinduló adatokat a további feldolgozáshoz.

Valamennyi multimédia-objektum esetében a digitális adatrögzítés eredménye az adott eszköz által generált szerkezetű állományként áll rendelkezésre. A gyakorlatban fontos kérdés, hogy a generált kimeneti formátumok a további feldolgozáshoz megfelelőek legyenek. Ellenkező esetben a következő munkafázist a szükséges konverziókkal kell kezdeni.

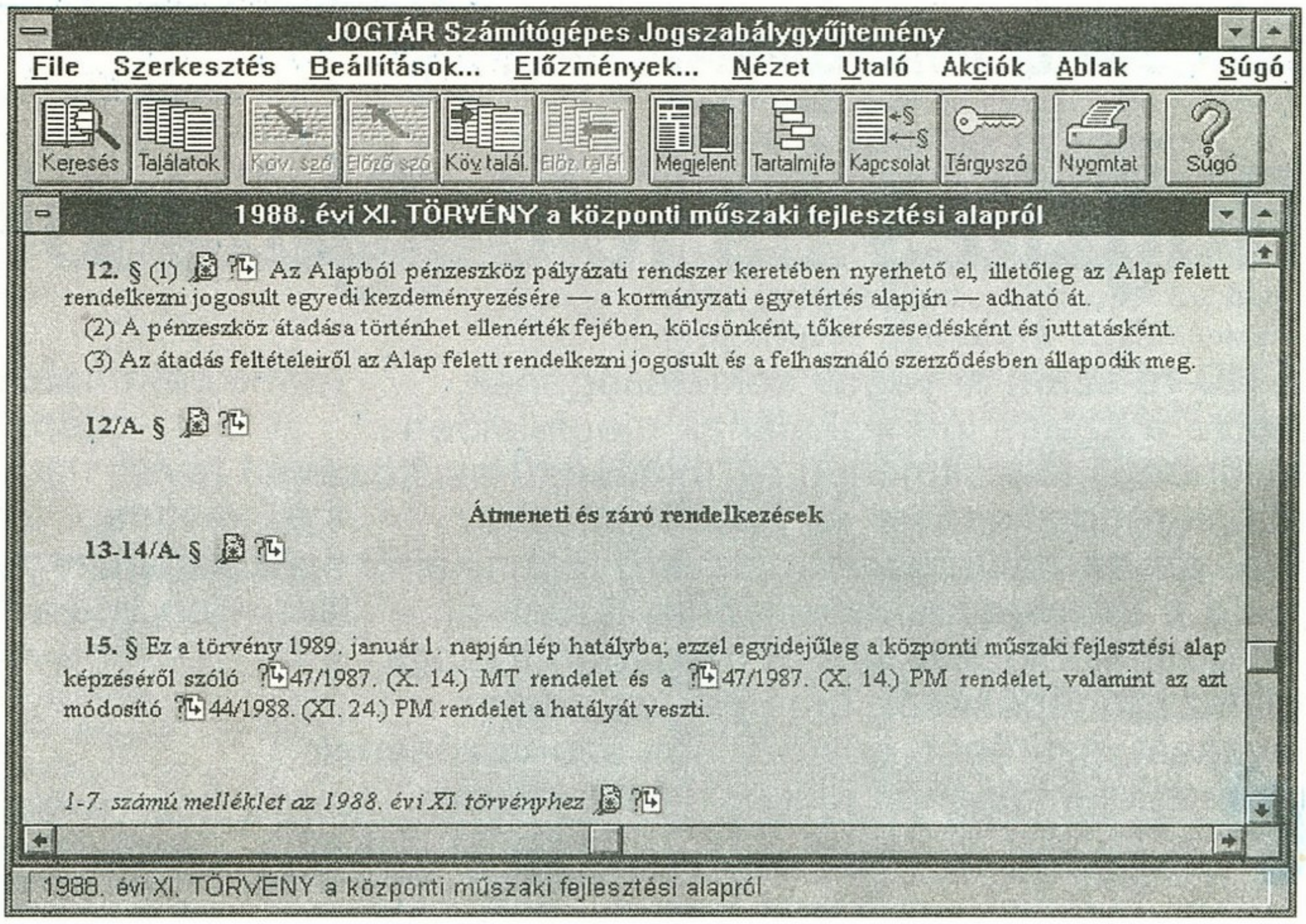
Az utalók cirkalmas társasága...

Az utalók rögzítése mehet kézzel vagy automatikusan, de már a további feldolgozás feladata a rögzített utalók adatbázisba rendezése.

Kézi feldolgozás esetén a szakember a már bevitt alapadatok megfelelő pontjait köti össze grafikus eszközök segítségével, vagy szövegekben az adatok közé elhelyezett speciális jelekkel.

Az automatikus utalófeldolgozás (utalógenerálás) akkor használható, ha az utalók kiindulópontja a forrásdokumentumban automatikus analízis útján felismerhető, és a végponthivatkozás szintaktikai analízissel meghatározható. A feldolgozó szoftver a felismert és generált utalókat közvetlenül az adatbázisba helyezheti.

Az automatikus utalógenerálás értelemszerűen az alapanyagban meglévő, formálisan felismerhető hivatkozások feldolgozására alkalmas. Bár ehhez esetenként egyedi analizátort (parsert) kell készíteni, azonban ennek elkészülte után az utalók előállítása igen hatékony és gyors. A kézi utalófelvétel ezzel szemben olyan információ bevitelét jelenti, amely az eredeti anyagban nem szerepel, vagy szemantikai kapcsolatot jelent. Ez a munka rendszerint a témát magas fokon ismerő szakember közreműködését igényli, ezért drága és időigényes.



Mikor kézi, mikor automatikus?

Automatikus utalófelismerés és -feldolgozás lehetséges például jogszabályokon belüli explicit kereszthivatkozások feloldására (amint az például a Jogtár Számítógépes Jogszabálygyűjteményben látható).

Hasonló példa a lexikonok szócikkszövegeiben található, más címszavakra való utalás. Az utaló szigorú formai jegyek alapján mindkét esetben felismerhető.

Ugyancsak automatikus utalófeldolgozás célszerű a képeken található, jól felismerhető szimbólumok, formák esetén. Elektronikus részegységek katalógusához a képek feldolgozásánál például a foltszerűen elhelyezkedő azonosítók lehetővé tették az ezeket lefedő érzékeny területek megtalálását, ezt követően az azonosító információ „elolvasását".

A fentiekkel ellentétben kézi utalózást igényel a jogszabálygyűjteményekben előforduló implicit hivatkozás. Sok esetben például a törvény egy később meghozandó rendeletre utal. Ilyenkor a rendelet megszületése után sem áll rendelkezésre a törvény megfelelő pontjáról a rendeletre mutató explicit formális hivatkozás, ezért az utaló felvitele csak a hatályosító jogász közreműködésével lehetséges.

Adatfeldolgozási funkciók

Konverzió. Az alaprögzítés során előállt adatok többféle szempontból igényelhetnek konverziót. Ez a folyamat automatikus eszközrendszer segítségével vihető végbe, a kézi beavatkozás minimális.

Formátumátalakítás. A különböző forrásból származó adatokat célszerű típusonként egységes formára hozni, annak érdekében, hogy az adatbázis előkészítésének, illetve az elkészült termékben való megjelenítésének feladatát megkönnyítsük. Ez formátumátalakítással jár, melyet az adott formátumokhoz illeszkedő szűrők (filterek) valósítanak meg. Ilyen filtereket a kereskedelemben kapható számos szöveg-, illetve képfeldolgozó szoftvercsomag tartalmaz.

Egységesítés. Az adatokat mind tartalmuk, mind formázásuk szerint egységesíteni kell. Ez magában foglalja az esetleges karakterkonverziókat, a fontkészletek, a formázó makrók, a palettakiosztások stb. egymáshoz igazítását. Az egységes részekből felépülő adatbázis kezelése problémamentes, megjelenítése kevesebb erőforrást igényel.

Az adatrögzítés során az adatok a munkaszervezésnek megfelelő időzítésben és sorrendben kerülnek felvitelre. Az adatbázis összeállítása előtt az információt a természetes sorrend (sajtóadatbázis esetén pl. időrend, szótáraknál alfabetikus rend stb.) szerint át kell rendezni. Problémát okozhat, ha több természetes sorrend is adódik. Ekkor a termékben leggyakrabban használt, vagy a legjobbnak ítélt felhasználást más módon lehetővé tevő sorrendet kell választani. A sorrendezést gyakran típusonként külön-külön kell elvégezni.

Strukturálás. Az (elsősorban szöveges) adatelemek belső szerkezetének kialakítása mind az adatbázis indexelése, mind a megjelenítés szempontjából fontos. Az adatrögzítés során az anyag rendszerint formázás és szerkezeti leírás nélkül lesz rögzítve. Ennek oka, hogy a nagy tömegű adatbevitel idő- és költségkeretei adatrögzítésre specializálódott személyzetet és technológiát kívánnak — nem többet. Az adatok visszakereséséhez szükséges különböző kategóriáknak (adatbázismezőknek) megfelelő indexek felépítése a dokumentum szerkezetének feldolgozásával lehetséges.

A szöveges információk szerkezetét reprezentáló hierarchikus címszerkezetet, a táblázathivatkozásokat és azok tartalmát, a kép és más multimédia-objektumok (hivatkozások) elhelyezését az erre felkészülteknek kell kialakítaniuk, többnyire kézi beavatkozással és ellenőrzéssel.

Összevonás, átcsoportosítás. Az előkészített adatokat gyakran többféle nézetben szeretnénk a CD-ROM-termékben megjeleníteni. E nézetek előállíthatók futásidőben, azonban ez néha időigényes, vagy nem a megfelelő minőségű eredményt adja. Ilyenkor célszerű a nézetekre már az adatelőkészítés során gondolni: adatok összevonásával, átrendezésével, kivonatolásával. Egy vektorformátumú adatokat tartalmazó adatbázis (tipikus példa erre a térkép) esetében általában többféle méretarányú megjelenítés szükséges. A legrészletesebb adatokat tartalmazó formából a termék megjelenítő programja elő tudja állítani a megfelelő nézeteket, azonban az ehhez szükséges algoritmus (apró részletek elhagyása, átskálázások stb.) időigényes.

A probléma megoldható, ha a termék az adatbázist többféle méretarányban tartalmazza, így a futásidejű skálázás elkerülhető vagy meggyorsítható. A többféle méretarányú adatok adatelőkészítési időben, hatékony algoritmussal, jó minőségben előállíthatók.

Matlák Tamás

Átrendeződő rendezvények

Melyik vásárra érdemes elmenni?

Magyarország két kiemelkedő számítástechnikai eseménye, az Ifabo és a Compfair szinte „kötelező gyakorlat" azoknak, akik gyors áttekintést szeretnének kapni a hazai kínálat valamelyik szegmenséről. A technikai újdonságok is nagyon hamar eljutnak e két kiállításra, mégsem tudják pótolni azt az élményt, amely a legnagyobb nemzetközi vásárokon — főleg a CeBIT-en és a Comdexen — vár a látogatókra. Ebben az évben nagy átrendeződés kezdődött az egymással versengő európai vásárvárosok között. Érdemes felfigyelni az új tendenciákra és törekvésekre.

Hannover Magyarországtól viszonylag messze van, nem könnyű odajutni, nem könnyű ott szállást szerezni... Mégis évente sok ezer magyar szakember zarándokol el a CeBIT-re, mert a világ legnagyobb számítástechnikai rendezvénye szinte kimeríthetetlen forrása az új ismereteknek, a szakmai kapcsolatteremtésnek. A nagyság persze átok is. Idén márciusban a látogatók száma már meghaladta a 750 ezer főt, a kiállítók száma a 6100-at, a nettó kiállítási terület a 317 ezer négyzetmétert. A CeBIT jelenlegi formájában elérte növekedésének határait. Régi vita az is, hogy kikhez szól a rendezvény: inkább az informatika hivatásos szakembereihez vagy a privát felhasználói szférához. E kettő szétválasztására tesznek most kísérletet. Az 1996-tól minden páros évben megrendezésre kerülő „CeBIT Home" az otthoni számítógéphasználat, a szórakoztató elektronika és a telekommunikáció teljes spektrumát felvonultatja majd, és a páratlan években sorra kerülő berlini Funkausstellung kiegészítő párjának lesz tekinthető. Az első CeBIT Home időpontja: 1996. augusztus 28–szeptember 1.

A márciusi „igazi" CeBIT tehát megpróbál még inkább a profi látogatókra koncentrálni. A „nem termelő" szféra leválasztása pedig egy kicsit a zsúfoltság csökkentését is szolgálja. Kérdéses azonban, hogy ez mennyire sikerül majd. A vásárkoncentráció belső törvényszerűségeit ugyanis nehéz befolyásolni. Miközben a nagyok egyre erősödnek, a kisebbek — még jelentős múlttal a hátuk mögött is — kidőlnek a sorból. 1995-ben lemondták Párizsban a Sicob, Birminghamben a Which Computer megrendezését. A legtöbb európai informatikai vásár mutatói is évről évre romlanak, miközben Hannover töretlenül fejlődik.

Nyugat-Európa főbb informatikai vásárainak legutóbbi mutatói*

Vásár (Év)	Város	Kiállítók száma	Terület (netto nm)	Látogatók száma
CeBIT (95)	Hannover	6.111	317.657	755.326
Systems (93)	München	1.767	56.229	118.935
Orgatec (94)	Köln	1.599	124.869	135.256
SMAU (94)	Milánó	911	60.303	238.940
Orbit (94)	Basel	806	31.509	70.001
Ifabo (94)	Bécs	562	31.439	73.818
Simo (94)	Madrid	422	21.413	111.229
Systec (94)	München	389	8.720	13.230
Data Office (93)	Stockholm	366	11.191	40.985
Efficiency Beurs (93)	Amszterdam	285	17.000	60.437
Bureau (94)	Brüsszel	250	16.000	48.065

*Forrás: Deutsche Messe AG. (Sorrendbe szedve a kiállító cégek száma alapján.)

A visszaesést átszervezéssel, továbbfejlesztéssel megállítani igyekvő vásárvárosok közül viszonylagos közelsége alapján számunkra München a legérdekesebb. Eddig kétévente volt a Systems, s közben a CAD/CAM-es Systec. Idén ősztől a Systems minden évben sorra kerül, a Systec pedig megszűnik, pontosabban tematikájával beleolvad a Systemsbe, melynek idei időpontja: 1995. október 16-20.

Új koncepciójának kialakításakor a Messe München GmbH eleve igyekezett kiküszöbölni a „termelési" és „fogyasztási" beállítottságú közönség közötti ellentmondást. A Systems a profikhoz szól és a számítástechnika valamennyi ágát felölelő szakmai tartalom jellemzi, beleértve a konferenciákat is. Egy hónappal később (november 24-26.) megrendezik viszont Münchenben a „bits & fun" című, árusítással egybekötött bemutatót, elsősorban az elektronika magánfelhasználási szempontok szerint összeállított új kínálatából.

A Systems rendezői a vásárösszevonáson kívül más ötletekkel is előálltak. Okulva a Las Vegas-i és a hannoveri multimédia-szekciók zajkavalkádjából, az egyik pavilont például Multimedia-City formájában, városszerűen rendezik be, ezzel szeparáltabb, egymást kevésbé zavaró helyszínt teremtenek a multimédia-eszközöknek. A kezdeményezést a szakma érdeklődéssel fogadta, és a meghirdetést követően két hónap alatt elkelt az erre a célra szánt 5000 négyzetméteres kiállítási terület.

Az idei Systemsen a Neumann János Számítógéptudományi Társaság szervezésében magyar nemzeti standot állítanak fel, kedvezményes részvételi feltételekkel. A bajor régió mindig vonzotta a magyar gazdaságot, a megújult Systems pedig felkeltheti a hazai számítástechnika vállalkozóinak érdeklődését. Ami pedig a vásárlátogatást illeti: München kocsival tényleg csak „egynapi járóföldre" van.

Végül a hónap témájához illő hír, hogy a Systems 95-öt Bill Gates fogja megnyitni. (De előbb, szeptember 1-jén még külön eljön Budapestre, a helyszínen dedikálni az Új Alaplap mostani számának Windows-összeállítását.)

Faklen Pál

Egy kongresszus és kiállítás ürügyén

Térinformatikai körkép

Júniusban zajlott le a Budapesti Műszaki Egyetemen a Nemzetközi Térinformatikai Kongresszus és Kiállítás. A földrajzi információs rendszerek (GIS = Geographic Information System) térhódítását a grafikus felhasználó felületek általánossá válása különösen felgyorsította. A GIS szakmai részleteiben most nem mélyedünk el (a jövő évben az egyik hónap témáját ennek szenteljük), itt csupán összeállítottunk egy csokrot az újdonságokról — főleg azoknak, akik már alkalmaznak ilyen rendszereket, vagy közel vannak ahhoz, hogy alkalmazókká váljanak.

Gemini-X a közigazgatásnak

A kiállításon a Budata Informatikai és Szolgáltató Kft bemutatta Gemini-X térinformatikai rendszerét, amely egy közigazgatási terület adatainak rögzítésére, megjelenítésére, naprakész nyilvántartására, kombinálására, kinyomtatására és archiválására szolgál. Magyarországon már 13 körzetben használják. A Budata az amerikai Integraph által kifejlesztett MicroStation CAD szoftverre alapozva készítette el felhasználói alkalmazásait, szöveges adatbázis-kezelőjét, és a szöveg és grafika összekapcsolását.

A Gemini-X programcsomag alkalmas személyi számítógépen szöveges és térképi adatbázisok együttes és különálló kezelésére. Jelenlegi adattartalma:

— Műszaki ingatlan-nyilvántartás.

— Földhivatali adatnyilvántartás (tulajdoni lap).

— Ingatlanvagyon-kataszter.

— Környezetvédelem.

A térképállomány metrikus pontossága az egységes országos térképrendszer földmérési alaptérképeihez előírt hibahatároknak megfelel, azaz a térképezett és a digitalizálás útján meghatározott koordináták közötti eltérés legfeljebb ± 0,2 mm.

Gradis-GIS

A DigiKom Kft elsődleges tevékenysége a német Straessle cég Gradis-GIS térinformatikai rendszerének magyarországi forgalmazása, a szoftver támogatása és szervize.

A Gradis-GIS munkaállomásra épülve együttesen kezeli és elemzi a közös relációs adatbankban tárolt geometriai és leíró adatokat. Az objektumorientáltan strukturált adatmodell komplex térbeli analízist tesz lehetővé, és a döntéshozatalban támogatja a városigazgatást, az energiagazdálkodást és a környezetvédelmet. Több felhasználó esetén a tárolt adatok hozzáférhetőségét, integritását és konzisztenciáját egy következetesen alkalmazott adatvezérlési koncepció garantálja a lokális hálózatokban.

A DigiKom, mint a Straessle magyarországi partnere forgalmazza a Gradis-2000 térbeli adatok digitalizálására és sokoldalú feldolgozására szolgáló interaktív grafikus rendszert, és a Gradis-Flotte járműbevetések tervezését, mozgásuk menedzselését, mobil egységek helymeghatározását, és mindezek képi megjelenítését lehetővé tevő térképalapú irányítórendszert.

Földtani Intézet

A Magyar Állami Földtani Intézet tevékenysége is térképi adatbázisok létrehozásához kapcsolódik, legyen szó akár területhasznosításról, akár környezetvédelemről, akár természeti források feltárásáról. Regionális környezeti problémák megoldására, önkormányzatok és közművek számára nyilvántartásokhoz adatbázisokat készítenek. Adatelemzési és kartográfiai célokat szolgáló tiszta vonalmű előállítására digitalizálási és adatfeldolgozási technológiákat dolgoznak ki. Adatfeldolgozásukat és adatbázisaikat elsősorban az Integraph eszközparkjára és az MGE szoftvermoduljaira építik. Szolgáltatásaik összefoglalása: térinformatikai adatbázisok létrehozása, felhasználóspecifikus környezetek kialakítása, hulladéklerakók helyének kijelölése, környezetvédelmi projektek és környezeti analízis, városi térinformatikai rendszerek, közműkataszterek, földtani térképezés, vízföldtani modellek, területrendezési és területhasznosítási tervek, regionális építésföldtani projektek és digitális adatbevitel.

Integraph

Az Integraph Magyarország Kft az anyavállalat 1995-ben kihozott, Intel processzorú számítógépekre alapozott, komplett térinformatikai programcsomagját, a Mapping Office-t mutatta be, amely az egyszer már említett MicroStation grafikus alapszoftverre épül, és mind DOS, mind Windows NT környezetben elérhető. Külön-külön is alkalmazható moduljai:

— MGE-PC — GIS-adatok bevitelére, nyilvántartására, karbantartására, tematikus térképezésre, lekérdezésre, zónakijelölésre és -manipulációra, valamint felszíni analízisre alkalmas ismert programcsomag Intel-adaptációja.

— I/RASC — szürkefokozatú és színes raszterképeket és térképeket importáló, megjelenítő képfeldolgozó rendszer.

— I/RASB — régi rajzok tisztítását, aktualizálását elvégző, bináris raszter adatokat képernyődigitalizálással vektorrá konvertáló hibrid raszter-vektor szerkesztő rendszer.

— I/GEOVEC — szkennelt térképi állományokat intelligens GIS vektoros formátumúvá átalakító konverziós eljárás.

Compumap atlasz

Közönséges személyi számítógépen teszi hozzáférhetővé Magyarország térképállományát a debreceni Politext Bt Compumap atlasza, amely jól használható az oktatásban, a tervezésben, a döntéselőkészítésben, az útvonaltervezésben. A rendszerhez tartozó szöveges adatbázis Magyarország legfontosabb

földrajzi objektumait és azok koordinátáit tartalmazza. A speciális eszközrendszer lehetővé teszi, hogy a felhasználó „saját" térképét is bevigye a 32 megadott, 1:200 000 arányú, 150 vagy 300 képpont felbontású, topográfiai térkép mellé. (Nagyléptékű politikai kereső és autós áttekintő Magyarország térkép is van mellettük.)

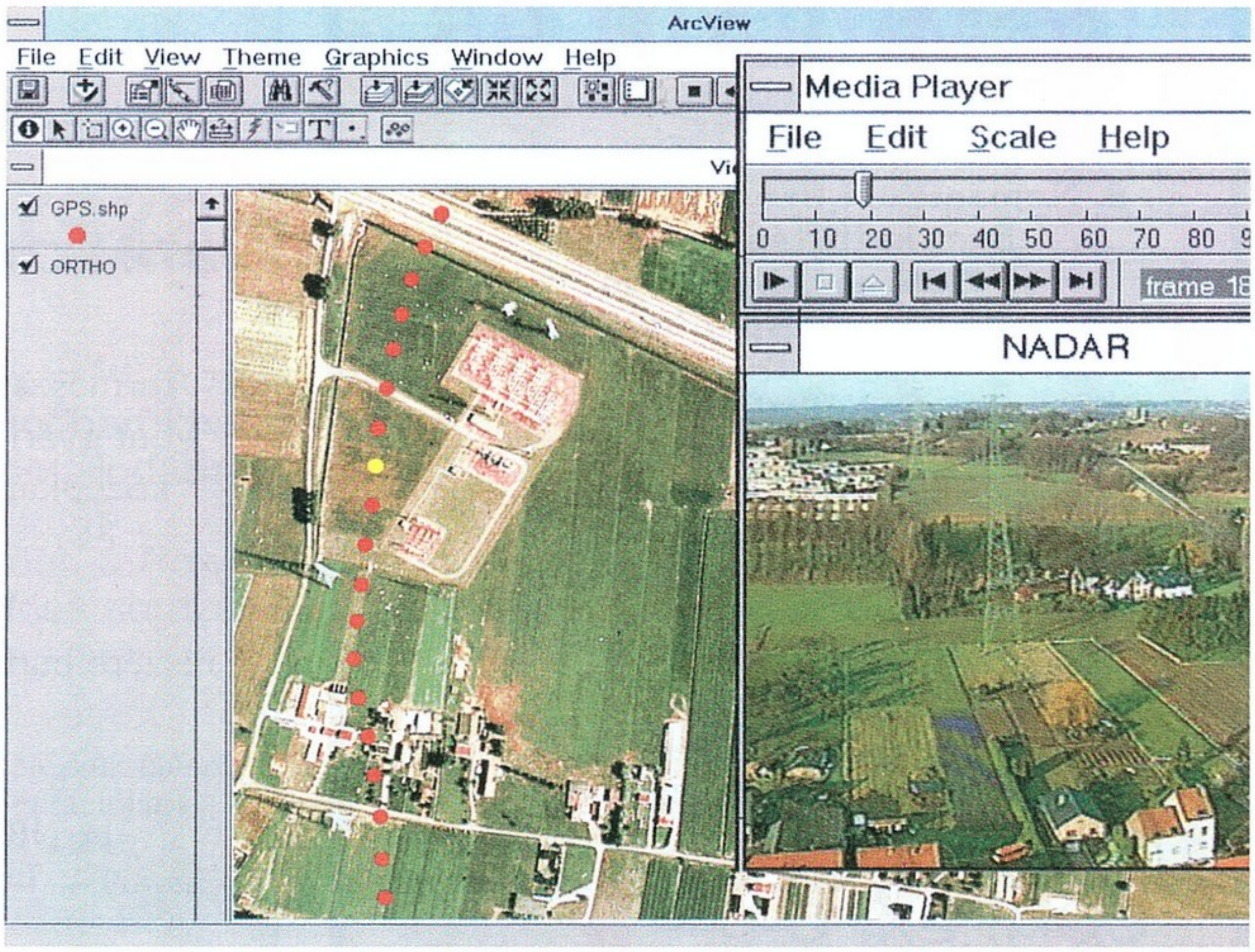
Alaptérképek

Bemutatkozott a székesfehérvári EFE FFFK Térinformatikai Tanszék, amely másfél éve alakult, és feladatának tekinti a térinformatika és a számítástechnika tárgyak egységes, információtechnológiai szemléletű oktatását, a térinformatikai gondolkozás fejlesztését, és a térinformatika szakmérnöki továbbképzés megszervezését.

Magyarország 7 szomszédos országgal közös határvonala meghaladja a 2200 kilométert, és ezen a vonalon több mint 50 000 töréspont található, melyek nyilvántartását a békeszerződések térképrendszereinek megfelelően a Földmérési és Távérzékelési Intézet végzi. De az ő feladatuk a nemzetgazdaság alapvető földmérési és térképészeti igényeinek kielégítése is, aminek érdekében biztosítani kell, hogy az ország teljes területére az egységes alappont-hálózat és az EOTR (Egységes Országos Térképrendszer) keretében földmérési alaptérképek és topográfiai térképek készüljenek. Intel lokális hálózati rendszeren üzemeltetik az ingatlan-nyilvántartás központi rendszerét, kidolgozňak digitális térképi termékszabványokat és hitelesítési technológiákat. Távérzékelési részlegük foglalkozik légifénykép- és műholdadatok beszerzésével, tárolásával, előfeldolgozásával, valamint alkalmazásorientált kutatási/fejlesztési projektekkel és nemzetközi technológiákkal a mezőgazdaság, az erdészet, a vízgazdálkodás, a térképészet, a környezeti és környezetvédelmi vizsgálatok területén.

Cartoranje

A Cartoranje Holland–Magyar Földmérési, Közműfelmérési és Általános Mérnöki Kft mint földmérő iroda vállal síkrajzi és magassági alappontsűrítést, pontmeghatározást, kataszteri munkálatokat, ingatlan-nyilvántartást érintő munkákat, közműfelmérést, közműalaptérkép-készítést, mérnökgeodéziai tevékenységeket, és olyan GIS-feladatokat, mint a térinformatikai digitális geográfiai adattárak előállítása, az alfanumerikus információk gyűjtése.



Légi video

Az Eurosense Nadar egy új szolgáltatás légi video-nyilvántartás létrehozásához GPS általi online georeferenciával, közvetlenül alkalmazható ArcView-val. Ajánlják karbantartási feladatokhoz, biztonsági rendszerekhez, kár- és katasztrófamonitoringhoz, környezetvédelmi analízishez, tájnyilvántartáshoz, nagyfeszültségű vezetékek melegedési pontjainak figyeléséhez, különböző infrastruktúrák tervezéséhez. A videofelvételeket helikopterről készítik Wescam típusú stabilizált, színes és infravörös, kettős érzékelőjű kamerával, ami tetszőleges irányból, bármilyen feltételek között nagy zoom intervallumú, éles és stabil képet biztosít. Az anyagot videolemezen CD-ROM-on vagy magnetooptikai lemezen tárolják. GPS (Global Positioning System — globális helymeghatározó rendszer) támogatja a légifényképezés navigálását. Az ArcView segítségével a GIS eddigi hagyományos földrajzi adatait egészítik ki a légi videofelvételekkel. A nagyvállalatnak nyolc országban van képviselete, köztük található a budapesti székhelyű Eurosense Kft is.

RD3

Az RD3 International teljes GIS-szolgáltatással jelentkezett, topográfiai térképeket, síktérképeket, digitális felszínmodellezést, digitális ortokromatikus fényképészetet és képek szkennelését magába foglaló fotogrammetriával, geodéziai irányításból, útvonalfelmérésből, kinematikai felmérésekből, határvonal-ellenőrzésből és GIS adattömörítésből álló GPS-szel, és adatbáziskiépítést, szoftverfejlesztést, rendszerintegrációt, hálózatszervezést, rendszerfelügyeletet jelentő GIS-szel. Bemutatták a telefonhálózatokhoz kifejlesztett, minden Unix platformon működő Degis szoftverüket, amely térképkészítésben, hálózatelemzésben, adatbevitelben és jelentéskészítésben támogatja a felhasználót. Az RD3 partnerei között egyaránt megtalálható például az USA hadügyminisztériuma, vagy a Monor Telefon Társaság.

Adatbázisra orientálva

Az 1969-ben alapított kaliforniai ESRI (Environmental Systems Research Institute, Inc.) a geográfiai információs rendszerek területén vezető szerepet játszik a világban. A földrajzi analízis és a térképészet számos fejlesztését tartja nyilván. Már kezdeti kutatásaikban is kartográfiai adatstruktúrákkal, speciális GIS szoftvereszközökkel, ezen rendszerek kreatív alkalmazásaival és automatizált térképkészítéssel foglalkoztak. Ma világszerte intézmények ezrei alkalmazzák az ESRI-termékeket, melyek között találhatjuk az ARC/INFO hatékony, flexibilis GIS-t, amely a világ első adatbázisorientált geográfiai információs rendszere volt, az ArcView desktop GIS-t és térképkészítő eszközt, vagy az ARC/INFO és az AutoCAD összekötését biztosító ArcCAD-et.

Timár István

Biztos alapok

Új IBM PC-korszak?

Tavaly decemberben az IBM-nek a PC-fronton várható lépéseiről még csak feltételes módban írhattunk. 1995. július 23-án a Kempinski Szállodában már Magyarországon is bejelentették az új személyiszámítógép-családot, a PowerPC processzorokra épülő Power Series sorozat első tagjait.

Az Apple—IBM—Motorola együttműködésből született PowerPC processzorok elődje az IBM RS/6000-es (RISC System/6000) munkaállomásaiban alkalmazott RISC architektúra. Az RS/6000 eredeti központi egységét 9 chip alkotja, ezek funkcióit fogja össze egyetlen integrált áramkörön a PowerPC processzor. Binárisan kompatibilis vele, így a PowerPC processzorokat az újabb RS/6000-esek is használják.

Már a tervezés kezdetén olyan család kialakítása volt a cél, amellyel széles teljesítménytartományt lehet átfogni. Ez azzal a nyilvánvaló előnnyel jár, hogy mivel egymással binárisan kompatibilisak, ugyanazt a szoftverbázist használhatják, s ez a felhasználó oldalán jelentős költségmegtakarítást eredményez.

A processzorcsalád jelenleg a következő tagokból áll:

— PowerPC 601 — olcsó asztali gépekbe szánt CPU. (Lásd a képen. ☛)

— PowerPC 603 — a 601-gyel azonos, kis fogyasztású, statikus működésű, hordozható gépekbe tervezett CPU.

— PowerPC 604 — nagyobb teljesítményű asztali munkaállomásokhoz készült.

— PowerPC 620 — hálózati szerverek és különösen nagy teljesítményű munkaállomások számára.

Az Apple lassan már egy éve árulja a 601-re épülő gépeit. Az IBM, úgy látszik, megvárta az erősebb típust.

Az új gépek

Két asztali és két hordozható gép alkotja a sorozat első részét. Az asztali gépek a Personal Computer Power Series (PCPS) 830 és 850 nevet kapták, a notebookok a cég sikeres ThinkPad gépeinek folytatását jelölve a ThinkPad Power Series 820 és 850 neveket.

Nézzük először az asztali gépek konfigurációját:

CPU:

— PowerPC 604: 100 MHz (Power Series 820).

— PowerPC 604: 100, 120 vagy 133 MHz (Power Series 850).

256 kbájt külső cache, ami 512 kbájtra bővíthető.

Memória:

16 Mbájt alap, 192 Mbájtig bővíthető szabványos SIMM modulokkal.

Bővítőbusz:

— A 820-ban 3 bővítőhely, 1 PCI és 2 ISA.

— A 850-ben 5 bővítőhely, 2 PCI és 3 ISA.

Video:

PCI-buszra csatlakozó SVGA, az S3 864 videovezérlővel, 2 Mbájt RAM-mal, de speciális 3D grafikára készült videoadapterekkel is rendelhető.

Háttértárak:

— 1,44 Mbájtos floppymeghajtó.

— 540 Mbájttól 1 Gbájtig terjedő merevlemez, a 820-ban 2, a 850-ben 3 meghajtóhely van.

— 4x sebességű CD-ROM beépítve.

Portok:

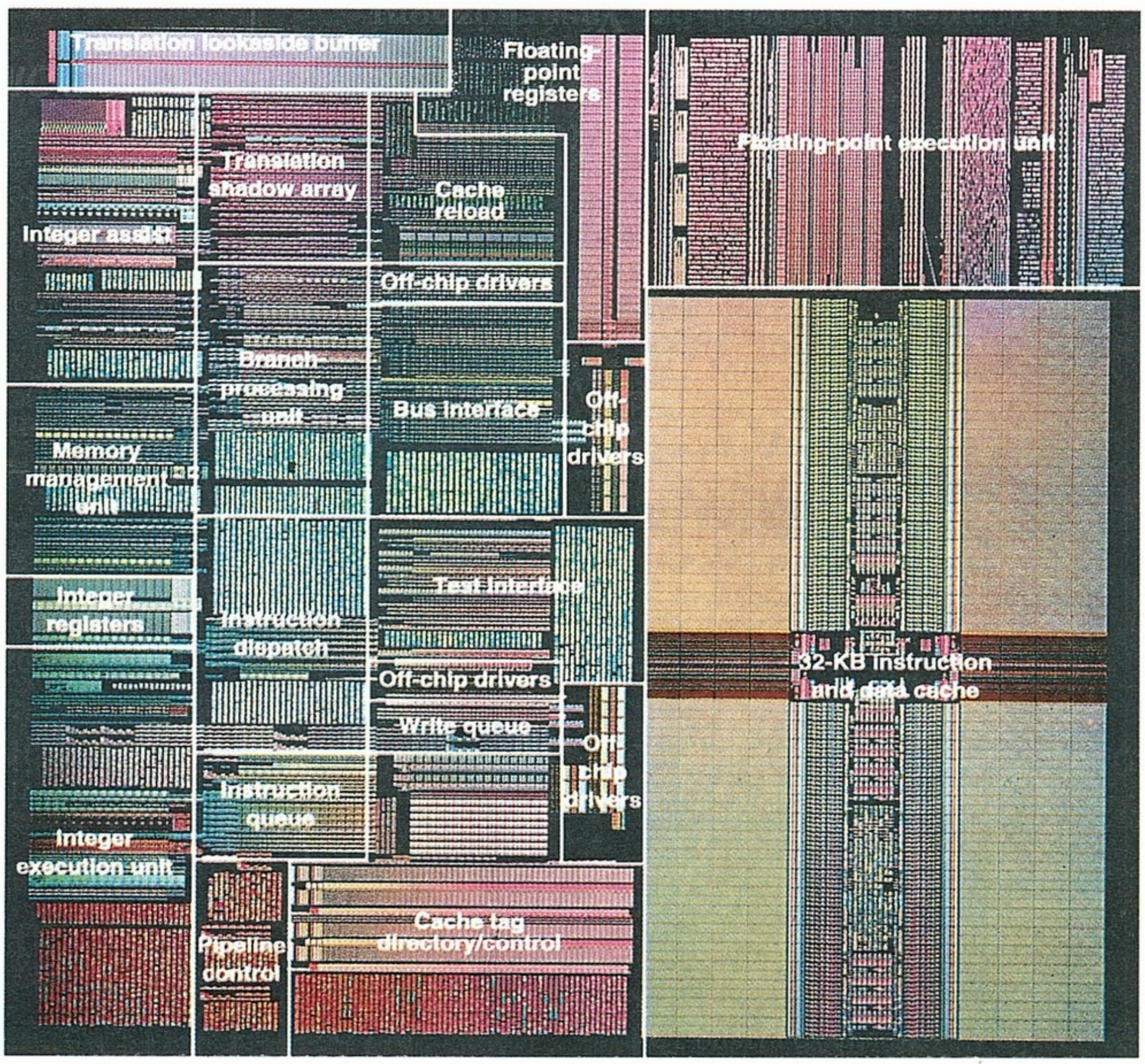
— Csavart érpárú, 10 Mbites Ethernet az alaplapon.

— 2 soros és 1 párhuzamos port, szintén az alaplapon.

Multimédia:

16 bites FM hangkártya, tartozék mikrofonnal.

A hordozható ThinkPad 820-as és 850-es gépek processzora 100 MHz-es 603, 256 kbájt külső cache-sel. A memória 16 Mbájttól 48, illetve 96 Mbájtig nőhet. A képernyő 10,4 inch átmérőjű aktív mátrixos színes LCD, 800x600-as SVGA vagy a normál VGA felbontással. A tárolók 1,44-es floppy, kivehető 540 Mbájt — 1,2 Gbájt kapacitású SCSI-2 interfészű harddiszk és beépített, dupla sebességű CD-ROM. Természetesen itt is van 16 bites audioegység mikrofonnal; és a notebookvilágban

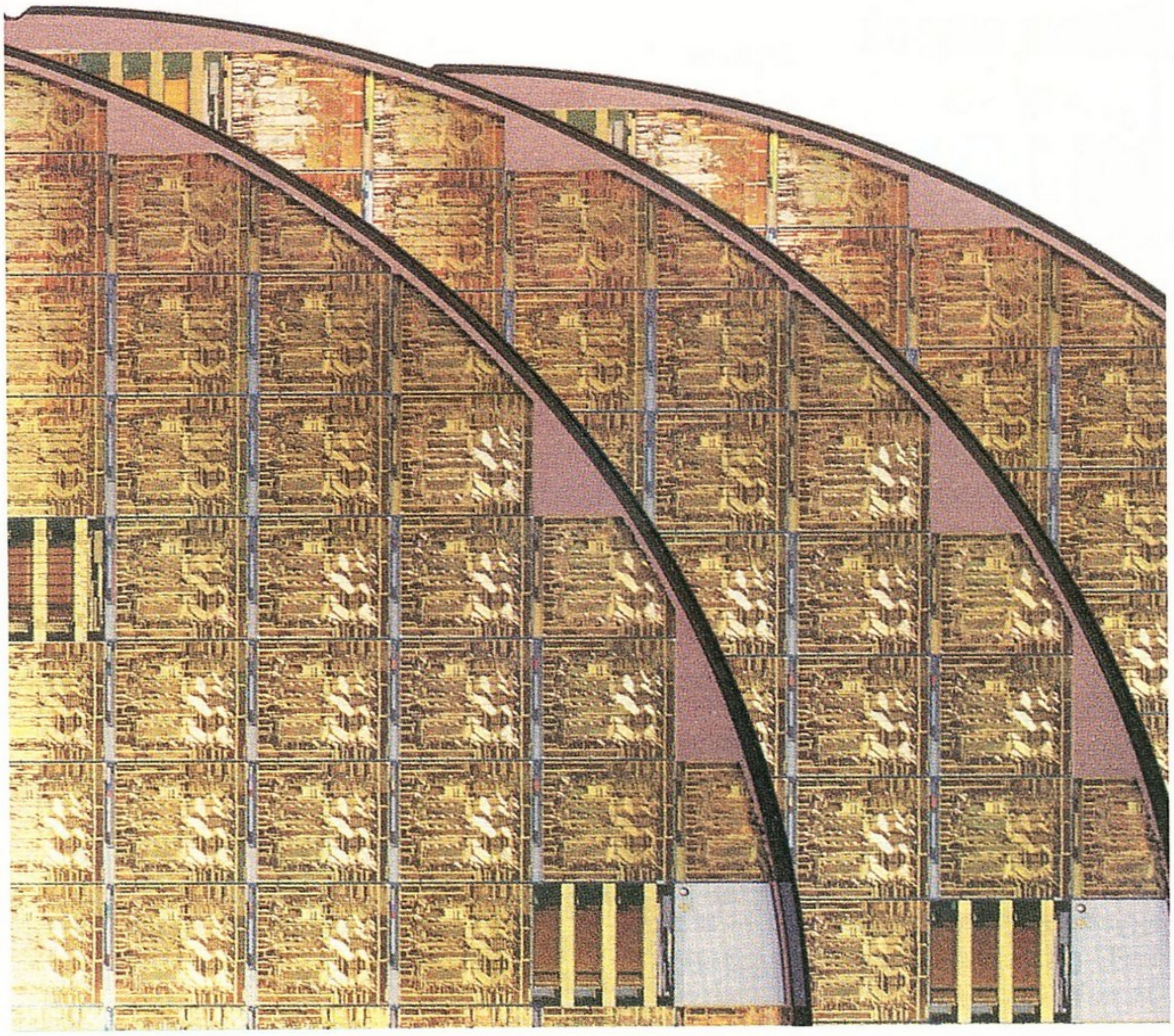


egyedülálló az, hogy a 850-es beépített videokamerával is rendelhető.

A bővítést PCMCIA-csatlakozó is lehetővé teszi, amelybe 1 db III-as vagy 2 db II-es típusú PCMCIA-kártya fér.

Mi fut rajta?

A gépeket bemutató Larry G. Smith szerint a hagyományos IBM PC/AT technológia kifulladóban van, és a Power Series most éppolyan továbbfejlődési lehetőség, mint egykor az IBM PC volt a házi játékgépekkel szemben. Természetesen az IBM sem gondolja, hogy a klasszikus PC-t belátható időn belül kiszoríthatja bármilyen más architektrúra, de választási lehetőséget akar adni a felhasználóknak.



A RISC processzorok teljesítménye és a gépek felszereltsége sokat ígérő, de megfelelő szoftverválaszték nélkül nem sokat ér. A Power Series sorozat binárisan kompatibilis az RS/6000-es családdal, azonban a munkaállomásokat — már csak az áruk miatt is — valamilyen célfeladatra, kiszolgálóként, mérnöki vagy tudományos tervezéshez, analízishez használják, ahol inkább a szofver minősége és nem a mennyisége a mérvadó.

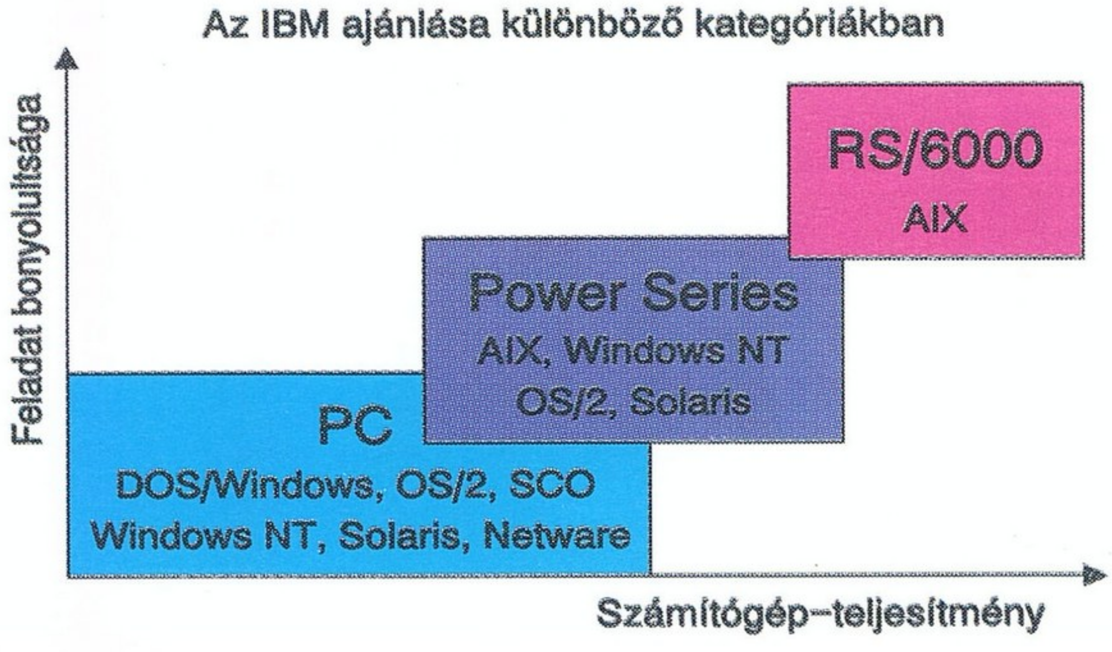
Amennyiben az IBM az „általános célú PC" kategória felhasználóit célozza meg, a gépekhez széles szoftverválasztékot is biztosítania kell. Nyilvánvaló, hogy a Power Series szofverháttérének megteremtése volt az egyik célja a tömegszoftverpiacon jól ismert Lotus felvásárlásának.

A Lotus táblázatkezelői és szövegszerkesztői eddig is a PC-s OS/2 piac több mint kétharmadát uralták; a cég megvétele abszolút logikus lépés az IBM részéről.

A Power Series PC-hez jelenleg egy 4 felhasználós AIX (az IBM Unixa) operációs rendszert és a Windwos NT PowerPC verzióját kínálják. Az év végére várható az OS/2 Warp és előbb-utóbb a Sun Solaris megjelenése a gépekhez. Az utóbbit Smith úr azzal indokolta, hogy szerinte három CPU vonal „marad életben": az Intel-sorozat, a PowerPC és (talán) a SunSPARC. Ebben valószínűleg nem értenek vele egyet a DEC Alpha-csapata vagy a Silicon Graphics Mips-fejlesztői.

A felhasználói szoftverek választékára tehát még várni kell, a kezdet mindenesetre biztató. Az Egyesült Államokban három héttel korábban jelentették be a gépeket, és ez idő során mintegy 15 ezer megrendelés érkezett.

A PowerPC 604 CPU-t használja az a hat új RS/6000 kiszolgáló és munkaállomás is, amelyet a Power Series gépekkel együtt jelentett be az IBM.



Új Warp

Számomra — sok jó tulajdonsága mellett — némi csalódást okozott, hogy az OS/2 Warpból hiányoznak a hálózati lehetőségek. Ezt pótolja a szintén most bejelentett OS/2 Warp Connect, amely a megszokott OS/2 Warp környezeten túl a következő lehetőségeket tartalmazza:

— Egyenrangú (peer-to-peer) hálózat kialakítása; nemcsak az OS/2 Warp Connectet, hanem a Windows for Workgroupst futtató felhasználókkal is.

— Kapcsolódás az IBM OS/2 LAN Serverhez és a Novell NetWare LAN-kiszolgálókhoz.

— TCP/IP protokoll a Unix rendszerekhez való kapcsolódáshoz.

— Távoli (remote) kapcsolódás modemen át másik PC-hez vagy LAN-kiszolgálóhoz.

A Bonus Pakon kívül az operációs rendszerrel kapjuk a Lotus Notes Express hálózati program kliens oldali részét.

Az OS/2 Warp Connect is két változatban kerül forgalomba, a Windows nélkül 229 $, a Windows kóddal együtt 299 $ áron.

Csórián Sándor

„Nyelviskola”

A Prolog mint következtető rendszer

Egyre inkább megkövetelik, hogy egy program intelligens legyen. Bemutatunk néhány olyan eszközt és módszert, amely ennek elérésében segíthet. Először megismerkedünk kevésbé ismert programnyelvekkel, amelyek a mesterséges intelligencia kutatása és eszközeinek kidolgozása kapcsán születtek, lettek érdekesek. Közben sok más dologról is megtudhatunk lényeges információkat, így lesz mibe „fogódzkodnunk”.

Az általános célú programnyelvekről be lehet bizonyítani, hogy egyik sem tud többet, mint a többi, azaz ha van egy programunk valamelyik programnyelvben, akkor ugyanazt a programot a többi nyelven is megírhatjuk. (Ezt szokás a programnyelvek Turing-ekvivalenciájának nevezni.) Ezért felesleges azon vitatkozni, hogy melyik nyelv jobb, melyik tud többet. Az viszont már érdekes, hogy mit milyen nyelven érdemes programozni.

A megszokott nyelveken (C, Pascal, Basic) azt programozzuk be a gép számára, hogy a program a végeredményt milyen úton érje el. Ekkor kézben tartjuk a program futását, viszont mindent aprólékosan le kell írnunk. Egy idő után ez igencsak unalmas lesz. Szerencsére vannak olyan nyelvek is, amelyeknél elegendő megadni azt, hogy mit szeretnénk megkapni, és még néhány alapvető tényt. Ha ezekből az adatokból elérhető a végeredmény, akkor a program automatikusan megkeresi azt az utat, mely a megoldáshoz elvezet. Az egyik ilyen nyelv a Prolog.

Pro Prolog

Mielőtt belekezdenénk tanulmányozásába, vegyük egy kicsit szemügyre a Prolog matematikai alapjait, talán ez majd segít a megértésben. A Prolog következtető rendszer. Az előre megadott állításokból próbálja meg levezetni a feltett kérdést. Azt, hogy egy állításhalmazból mikor vezethető le más egyéb állítás, a matematikai logika tárgyalja. Már igen régen kiderült, hogy az ilyen feladatok között vannak olyanok is, amelyeket nem tudunk megoldani algoritmussal (illetve programmal).

Ha nem használnánk változókat, akkor ilyen problémánk nem lenne, viszont ezzel a kérdések körét is igen leszűkítenénk. Ezért egy köztes megoldást választottak, mert mind az előre megadott adatokra, tényekre, szabályokra, mind a kérdésekre vannak bizonyos megkötések. Ha változót használunk a kérdésben, akkor a kérdés csak olyan lehet, hogy van-e olyan objektum, amelyre igaz, hogy... Például: Van-e olyan ember, akinek 'János bá a nagyapja', és 'van egy lánya'? A előre megadott tények, szabályok pedig általános érvényűek, azaz minden objektum esetén igazak.

Ténynek nevezzük azt, ha az állítás atomi, azaz tovább nem osztható, így például „a tej fehér”, „Feri beteg”, „2 páros”. A szabály pedig úgy néz ki, hogy bizonyos feltételek együttes teljesülése esetén teljesül valami újabb állítás is, így például „ha '6 óra van', és 'józan a harangozó', akkor 'harangoznak'”, vagy egy másik példa: „ha 'XY köhög', és 'XY lázas', akkor 'XY beteg'”. Ez utóbbi lényegében egy olyan séma, amelyben a változókat valami objektumra cserélve (például XY-t Pistára) újabb állításokat kapunk.

A példákban láthatjuk az 'és'-t, de a 'vagy'-ot nem. A Prologban a 'vagy' olyan, mint a goto a Pascalban, azaz van, de ha lehet, nem használjuk. Azt a módszert, amellyel a Prolog a következtetéseket megkeresi, rezolúciónak nevezzük, a lényege az, hogy olyan változókat találjunk, amelyek majd megfelelnek céljainknak. Mivel előre nem tudjuk, mire is lesz szükségünk, gyakran jutunk zsákutcába, ebből a visszalépés (backtrack) segít ki.

Prolog-alapok

Most már kezdődhet a „gyorstalpaló”. Az azonosítók a Prologban általában betűvel kezdődnek, és betűkkel, számokkal vagy aláhúzásokkal folytatódhatnak. Ebben a nyelvben fontos megkülönböztetni a kis- és nagybetűket, mivel a változókat a kezdő nagybetű jelöli. Ezért kell majd nemecsek_ernő stílusban írni a személyek nevét, nehogy ezeket az azonosítókat változónak tekintse a program. Minden más azonosítót és az aposztrófok közötti karaktersorozatot atomnak nevezünk.

A lista fogalmáról (ami igencsak alapvető a Prologban) még sokat fogunk beszélni. Egyelőre legyen elég annyi, hogy a listát szögletes zárójellel jelöljük, az elemeit pedig veszővel választjuk el. Lista eleme lehet atom, változó, de akár egy újabb lista is. A listának az első elemét (mint speciális elemet) a többitől egy függőleges vonallal választhatjuk el. A Prolog programokat két részre oszthatjuk, egy adatbázisra és egy tudásbázisra. Az adatbázist tények alkotják, mint például *szeret(ádám, éva)*. (A Prolog nem szereti az ékezetes betűket, de mivel a példák így olvashatóbbak, én csak így írom.)

A Prolog a pontot használja zárójelként, azaz innen tudja, hogy egy állítás vagy kérdés feldolgozásához hozzákezdhet. Ezért nyugodtan használhatjuk a soremelést szabályok megadása közben. A tudásbázis elemei (a szabályok) a következő formájúak:

szeret(ádám, éva):– szőke(éva), csinos(éva).,

ahol el0l szerepel a szabály feje, a :– jel után pedig a szabály törzse.

Ha már van egy Prolog programunk, akkor annak kérdéseket tehetünk fel. Ha a program csak az előbb említett tényt tartalmazza, akkor a *?– szeret(ádám, éva)*. kérdésre *yes*-t válaszol, míg a *szeret(ádám, lucifer)*. kérdésre *no* a válasz. Ha a tények között nincs meg a megfelelő állítás, akkor a tudásbázis-

ban keres egy olyan szabályt, amelynek a kérdés a feje, és ezen szabály törzsét tekinti új kérdéseknek, és most már ezekre keres sorra választ. Ha mindegyikre *yes*-t kap, akkor a válasza is *yes* lesz, ellenkező esetben újabb szabállyal próbálkozik. Ha már nincs több lehetőség, csak akkor válaszol *no*-t.

Hasonlóképp megy ez változókkal is. Nézzük azt a példát, amely egyeseknek már nagyon ismerős lehet:

nagyapa(X,Z):–apa(X,Y), apa(Y,Z)., apa(béla,dénes.), apa(ede, béla), apa(ede, géza.

A nagyapa kapcsolatot egy általános szabállyal definiáltuk, amelyről már tudjuk: igaz lesz a változók tetszőleges helyettesítése esetén is. Ha kérdésben szerepel változó, ekkor a program nemcsak igennel válaszol, hanem megadja azt az értéket is, amelynél az állítás teljesül. Legyen a kérdés *?–nagyapa(béla, A).*! Ekkor, mivel a kérdés a nagyapa szót tartalmazza, az első szabállyal kell foglalkozni, és megfelelő X,Y,Z -t kell választani. Az X értéke adódik a kérdésből, de Y-t és Z-t még meg kell keresni. Rövid idő alatt a gép is rájön, hogy ilyen Y és Z nem létezik.

Tömör és rugalmas

A Prologban igen rövid programokkal lehet megoldani majdnem minden dolgot. Egy bináris fát jelöljünk a következő módon *[bal_fa, érték, jobb_fa]*, ahol a *bal_fa* és a *jobb_fa* a megfelelő részfák, azaz újabb listák. Ha már egy levélnél járunk, mivel annak részfái nem léteznek, ezért ezeket *[]*-val, azaz üres listával jelöljük. Nézzük azt a programot, amely egy új elemet szúr be a már rendezett bináris fába, és ezt a programot jelöljük *beszúr(érték, fa, új-fa)* formában (feltesszük, ez az új érték még nincs a fában). Ha a fa üres, akkor ezt az értéket kell beírni a gyökérbe: *beszúr(X,[],[[],X,[]]).*, míg ha nem üres, akkor a gyökér értékétől függően kell a bal, illetve jobb oldali részfába elhelyezni:

beszúr(X, [bal_fa,Y,jobb_fa], [bal_fa1,Y,jobb_fa]) :– XY, beszúr(X, bal_fa, bal_fa1).,

valamint

beszúr(X, [bal_fa,Y,jobb_fa], [bal_fa,Y,jobb_fa1]) :– XY, beszúr(X, jobb_fa, jobb_fa1)..

Mint láttuk, rekurzívan oldottuk meg a problémát, és ami lényeges, ez a program önmagát csak az utolsó lépésben és csak egyszer hívta meg. Ez nagyon speciális, mert ilyenkor a rekurzióból lényegében iteráció (ciklus) lesz, és épp ezért melegen ajánlott. Ezeknek a rövid programoknak az előnye az, hogy könnyen meg lehet érteni, és könnyű hibátlanul megírni őket. (Ha valamiért mégis hibás, gyorsan ki lehet javítani, és az újabb változatokat is pillanatok alatt ki lehet próbálni. Ugyanaz a növekményes programozás jellemző a Prologra is, amely a Forth erejét is adja.) Ha a végső program sebessége megfelelő, akkor a részprogramjainkat felesleges bántani; de ha valamiért nagyon lassú a programunk, amin egyáltalán nem kell csodálkozni, akkor csak azokat a programrészeket kell kicserélni, amelyek ezért a sebességért felelősek.

Kettesével...

Talán mindenki ismeri a Fibonacci-számokat. Ez egy sorozat, ahol az első két szám egyes, míg minden későbbi az előző kettő összege, azaz a sorozat valahogy így kezdődik: 1, 1, 2, 3, 5, 8, 13, ... Könnyedén írhatunk e definíció alapján egy Prolog programot, mely kiszámolja a sorozat n-edik tagját, például a *fib(n,x)*-ben n jelölje a sorszámot, x pedig az n-edik Fibonacci számot. Ekkor a két alapvető tény a *fib(1,1).* és a *fib(2,1).*. A többi számot pedig a képzési szabállyal kapjuk:

fib(n,x) :– n2, n1 is n–1, n2 is n–2, fib(n1,x1), fib(n2,x2), x is x1+x2..

Ebben a példában szerepel az *is* szócska, ami az értékadásnak felel meg. Ez a mögötte szereplő kifejezés értékét adja az előtte szereplő változónak. Remélem, így már érthető ez a példa. Ha egy nagyobb n esetén akarjuk kiszámolni a sorozat n-edik tagját, akkor a sorozat korábbi tagjait ezzel a módszerrel igen sokszor feleslegesen számoljuk ki, ezért is lesz a program futásideje exponenciális.

Ha tároljuk a menet közben kiszámolt értékeket, akkor már lineáris idő is elég a megoldáshoz. Mivel csupán két értéket kell folyamatosan észben tartani, az is lehet egy megoldása a linearizálásnak, hogy a programfejbe újabb változókat építünk be, amelyek ezeket az észben tartandó értékeket tartalmazzák, és a kisebb értékektől haladunk a nagyobbak felé.

Egy másik megoldás lehet az, hogy listában tároljuk a kiszámolt értéket, vagy akár ki is bővíthetjük a tudásbázist a már megismert Fibonacci-számokkal. Aki egy kicsit járatos az analízisben, az persze alkalmazhatja a megfelelő zárt formulát, azaz konstans idő is elég a megoldáshoz. Ez talán jó példa arra is, hogy Prologban könnyen lehet olyan programot írni, amely egyáltalán nem hatékony, valamint arra is, hogy ugyanazt igen sok módon meg lehet írni.

Személyes meglátások

Sajnos az *is* szócska, vagy az adatbázis manipulálása nem illeszkedik a Prolog alapfilozófiájához, mert például az első példabeli *szeret* esetén a kérdésben használhatunk tetszőleges argumentum helyén változót (azaz rákérdezhetünk arra is, hogy kit szeret 'ádám', de arra is, hogy kit szeret 'éva'); míg az *is* esetén megengedett a *?– X is 2+3.* kérdés, amelyre a válasz természetesen 5 lesz, de az *?– 3 is 2+X.* kérdésre hibajelzés a válasz. A Prolog tervezői szerintem túl nagy szerepet szántak ezeknek a procedurális részeknek. (Nemrégiben szétnéztem a jelenleg kapható Prolog rendszerek között, és már majdnem mindegyikben megvan az a lehetőség, hogy a Prolog programok más nyelven írt függvényeket hívhassanak meg, illetve más nyelvű programok használhassanak Prolog programrészeket. Ez a másik nyelv általában a C, de néha más is előfordul, mint például a Visual Basic. Az ilyen „piszkos trükköket" nyugodtan ki lehetett volna adni ezeknek a nyelveknek.)

Ha már itt tartunk, érdemes leírni, hogyan is szereztem tudomást e programok létezéséről. Ehhez nem elég bemenni a sarki számítógépes boltba, mert a hazai üzletek nem igazán vannak tele programokkal. Van bennük pár jelenleg divatos szoftver, de ha másra van az embernek szüksége, vagy ha csak kíváncsi rá, hogy a hasonló programok nem olcsóbbak-e, akkor már kutatómunkát kell végezni. Hol érdemes elkezdeni a kutatást? Ha van Internet-kapcsolatunk, akkor egyértelműen ezen; ha viszont nincs, akkor ki kell osztani a feladatot valaki olyanra, akinek van.

Először bele kell nézni a Useneten a NetNewsba, van-e valami a Prologról? Szerencsére egy egész NetNews-csoport foglalkozik ezzel a témakörrel: a comp.lang.prolog. Itt ki kell várni a hónap közepét, mert mindig tizenötödikén jön le a FAQ. Ez nekem már jó kiindulási alapot adott, mert egyrészt majdnem húsz, PC-n futtatható kereskedelmi Prolog-verziót említett meg a fejlesztők címével, telefonszámával és (ami a legkényelmesebb:) e-mail-címével. Egy körlevelet írtam mindegyiküknek, de csak hat cég tartotta érdemesnek válaszolni egy kis ország egyik egyetemének. Innen már nem volt nehéz kiválasztani a megvásárolandó programot. (Nem csinálok reklámot, nem írom

meg, hogy választásom melyik programra esett.)

További tanulságok

A megküldött ismertetőkből egy-két érdekesség kiolvasható: egyik gyártó sem tud megszabadulni a Microsoft hegemóniájától, vagyis mindegyik programnak van windowsos verziója. Az egyik ismertető igen hosszú listát adott, amelyben azok a cégek szerepelnek, amelyek ezt a bizonyos Prolog programot, vagy az ebben a környezetben fejlesztett programokat használják. E listát végigolvasva rájön az ember, hogy nem csak C++-ban programoznak, és igen sok alkalmazásban nagyon jól kihasználhatók a Prolog lehetőségei.

Minden nyelv bizonyos szempontból nehézkes, azaz vannak olyan részei, amelyeket könnyebben lehet valamelyik más nyelven megfogalmazni. Manapság már sok olyan rendszer van, amely elősegíti a többnyelvű programozást, ám ha a programunkat hordozhatóvá akarjuk tenni, akkor minden nyelvből csak azt a részt használhatjuk fel, amely minden interpretációban megvan, azaz többé-kevésbé szabványos. Nem mindig lehet előre látni, hogy milyen operációs rendszerre kell átültetni programunkat, és akkor ott majd milyen fordítók lesznek elérhetők. Ezért megoldás lehet kedvenc (és bárhol használható) nyelvünkön megírni egy interpretert, amely a többi nyelvű részt képes lefordítani és futtatni. Pontosabban szólva az általunk használt nyelvet ki kell bővíteni a számunkra szükséges más nyelvi elemekkel.

Ilyen rendszerre jó példa az Until és a SLang, ahol mindkettő C nyelven megírt Forth, illetve szövegkezelői rendszer, amelyet tetszőleges C programba beépíthetünk. Ehhez a cikkhez illeszkedve a Prolog megvalósításait (vagy legalábbis annak egy részét) kellene felsorolni. (Ezt a felsorolást most mellőzöm, de aki kíváncsi rá, az a Prolog FAQ-ban mindent megtalál.) Elvétve létezik egy-két C nyelvű program és nagyon sok Lisp, illetve Scheme gazdanyelvű.

A kíváncsiak kedvéért

Ha valaki még többet akar megtudni erről a nyelvről és a mesterséges intelligenciáról, akkor tanácsolom: ai+query@cs.cmu.edu e-mail-címre írjon egy levelet, amelyben csupán egy help szócska szerepel. Olyan levelet kap válaszul, amelyben szerepelnek mindazon parancsok, amelyeket ez a gép megért. (Ezekkel a parancsokkal elküldhetjük az ide tartozó FAQ-kat, keresgélhetünk a mesterséges intelligencia adat- és programbázisában.)

Ha kiválasztottunk valamilyen fájlt, akkor azt leszedhetjük az ftp-vel is, de szerintem ez tikkasztóan lassú, ezért mindenkinek inkább a mailftp-t javaslom. Ehhez érdemes a bitftp@pucc.bitnet címet használni. Ide is egy help tartalmú levelet érdemes küldeni először. Ebben a programbázisban PC-re legalább öt különböző közkincs Prolog-környezet létezik. (Kereskedelmi verzióban ennek a duplája.) E közprogramok egy része még egy XT-n is eldöcög, viszont vannak olyanok is, amelyeket csak Windows alól indíthatunk el. Ugyanitt találhatunk Mike néven egy Prologban megírt szakértői keretprogramot, amelyet akár saját tudásunkkal is feltölthetünk, vagy használhatjuk a mellékelt tudásbázisokat. (Ez a program megtalálható a mostani lemezmellékleten.)

Ezt a programot azért is érdemes átfutni, mert igen sok ismeretet meríthetünk a forrásból. Sok más példaprogramot találhatunk itt is, de más gépeken bőven van Prologgal kapcsolatos anyag, ezek megkeresésében segíthet a FAQ vagy archie.

Mivel a Prolognak folyamatosan keresnie kell a megfelelő változókat és állításrészeket, így mindenképpen tartalmaz egy adatbáziskezelőt, amely a kereséseket automatikusan elvégzi. A Prolog nyelv megvalósításai mindig az első ténynél, illetve szabálynál kezdik a keresést, ezért néha ügyelni kell a megfelelő sorrendre. Aki már ismeri saját Prolog rendszerét, jól lerövidítheti programjait azzal, hogy kihasználja ezt a sorrendet a szabályok leírásában, viszont ez azzal a hátránnyal járhat, hogy azt a Prolog programot már nem párhuzamosíthatjuk. Pedig léteznek (igaz, nem DOS alá írt) párhuzamos Prolog rendszerek is, és ezek között még közprogramok is vannak.

A Prolog interpreterhez már csak a változók megfelelő megválasztását kellene programozni. Erre van két egyszerű, ám kicsit hosszú algoritmus. Sőt időigényes is ez az eljárás, így fordult elő az, hogy a sebesség növelése miatt egy fontos szabályt kihagytak, azaz van olyan eset, amikor a Prolog „a szemünkbe hazudik"! Vannak próbálkozások, hogy ezt a hibát valahogy kivédjék, de egyik sem az igazi. Kellő körültekintéssel kezelve a dolgot, valamint a háttér alapos ismerete mellett nem tud átrázni senkit sem a rendszer.

Középhaladóknak

Aki már úgy érzi, hogy otthonos a Prologban, próbáljon megírni (vagy csak megtervezni) egy órarendkészítő programot. Majdnem erre a feladatra találták ki Prologot, mert ez a probléma is megfogalmazható egyetlen szabályrendszerrel. (Hogy élethű és használható legyen a program, olyan szabályokat is be kellene tenni, hogy Szabó tanár úrnak ne legyen nulladik órája, mert úgysem jönne be rá; és Kovács tanárnőnek délutánra ne rakjanak órát, mert megy a gyerekért a bölcsődébe.) Egy probléma van ezzel a programmal, hogy igen tekintélyes adathalmazon dolgozik, és ha nem az optimális sorrendben adjuk meg a tényeket és a szabályokat, akkor esetleg több hétig tart a megoldás egy nagyon gyors gépen is. (Ilyenkor az sokat segít rajtunk, ha a feladatot egymástól független részfeladatokra tudjuk bontani.)

Hasonlóan érdekes feladat lehet megírni egy olyan Prolog programot, mely önállóan kirakja a bűvös kockát. (Ilyen program már létezik, egy nem különösebben elterjedt Prolog dialektusban írták meg, de szerencsére mellette megtalálható a PC-n futó EXE fájl is.)

Hamarosan a Lispet és a Scheme-et veszézzük ki.

Aszalós László

KÖVETKEZŐ SZÁMUNKBAN A HÓNAP TÉMÁJA:

NYOMTATECHNIKA

Tanulni nehéz — de valahogy kell

A rendszer intelligenciája

A rovat legutóbbi cikkeiből leszűrhettük: az intelligens számítógépes tervezőrendszerek feladata a mérnökök tervezési tevékenységének reprodukálása. Mivel ma még nem tudunk elképzelni hatékonyabb formát az emberi végrehajtásnál, a reprodukálás tulajdonképpen csak jobb vagy rosszabb lemásolása annak, amit az ember végez. Napjainkra a „hagyományos" szemléletű szakértőrendszerek lehetőségei (és természetesen a korlátai) ismertté váltak. A nagyobb hatékonyság, a jobb alkalmazkodás és az alternatívák kezelésének lehetősége érdekében a kutatóknak kifinomultabb tudásfeldolgozási formákat kell a tervezőrendszerekbe építeniük. Az alábbiakban az említett célok eléréséhez egyik legfontosabbnak ítélt feltétellel, a számítógépi tanulással foglalkozunk.

A szakértőrendszerek elvét követve megvalósított tervezőrendszerekkel kapcsolatban három örökletes korlátozást lehet megemlíteni. Az egyik a „zárt világ effektus", ami azt jelenti, hogy a következtetés és a tervezési döntéshozatal csak az adott időpontban explicit (kifejtett) formában rendelkezésre álló tudásbázis-tartalom alapján történhet. A másik az egyszintű következtetés problémája, amely nem teszi lehetővé absztrakciók vagy leszármaztatások végrehajtását. A harmadik az adaptivitási készség hiánya, amely a rendszer ismeretkészletének és problémamegoldási módszerének (esetleg működési folyamatának) a konkrét esetekhez való igazítását tenné lehetővé.

Ha e három hiányosság okát keressük, nem nehéz megállapítani, hogy valamennyi közös forrásra vezethető vissza. Ez pedig a tanulási képesség, amely sok tekintetben az ember egyik leginkább csodálatra méltó, de fiziológiája és pszichológiája tekintetében még közel sem feltárt képessége. A cikk írója más szakértőkkel egyetértésben úgy gondolja, hogy a tanulás beépülése lehet a kulcs a tervezői problémamegoldásban nélkülözhetetlen nyitottsághoz, absztraháláshoz és alkalmazkodáshoz.

Tanulás nélkül a tervezőrendszerek csak olyan szituációkkal képesek megbirkózni, amelyekre vonatkozólag explicit tudással rendelkeznek. A tanulási képesség viszont lehetővé teszi a rendszereknek, hogy alkalmazkodjanak a megváltozott szituációkhoz.

Nyitottság kell

Az ember esetében a tanulás a lexikális ismeretek, gyakorlati készségek, feladatmegoldásban való jártasságok, meghatározott viselkedésfajták elsajátítását jelenti. Az ember a környezetével való szoros kapcsolata alapján tesz szert új ismeretekre, vagyis a tudás bővíthetőségének egyik feltétele a nyitottság. Mivel a szakértőrendszerekre oly sokszor eredendően jellemző „zárt világ szindróma" ezzel ellentétben áll, feloldását a fejlett tervezőrendszerekben kell elérni.

A pszichológiából ismerjük, hogy Ebbinghaus és Binet az emberi tanulást az emlékezéssel hozta szoros kapcsolatba. Az emberi agy asszociatív tárolóként való értelmezése a tanulást új reflexkapcsolatok létrehozásának folyamataként állítja be. Az emberi tanulást olyan mozzanatok fedik le, mint az észlelés és tapasztalás, a felfogás és megértés, a korábbi elsajátítások eredményeivel való kapcsolatba hozás, a megőrzés és felidézés.

Az emberi tanulási ciklus egyaránt jelentkezik a tervezésben rövid távon (azaz a megoldási alternatívák keresése során) és hosszú távon (azaz a múltbeli tapasztalatok felidézése kapcsán). Másrészt, és ez összhangban áll korábbi meghatározásunkkal, a tanulási ciklusok során a tervezők nemcsak ismeretkészletüket bővítik, hanem javul a kreativitásuk is. Minderre azért tértünk ki, mert fontos látni, hogy mi a szerepe a tanulásnak a tervezésben.

Nem kiselefánt...

Amikor egy tervezőrendszer irányított problémamegoldást hajt végre, a tudásbázis tartalmának bővítése mellett meg kell szereznie azt a képességet is, hogy ugyanolyan problémát a későbbiekben nagyobb hatékonysággal és igényesebben oldjon meg. Ez az, amit az ilyen rendszerektől el lehet várni. A tanulás által megalapozott adaptálódási képesség feltételezi a strukturális változásokat, de a rendszer vezérlésének módosulását is.

Figyelembe véve, hogy az összes lehetséges tervezői szituáció kezeléséhez szükséges aprólékos tervezői tudást — idő- és erőforrás-igényessége miatt — gyakorlatilag nem lehet formába önteni és feldolgozni. Ezt helyettesítheti, ha a rendszer az alapvető fizikai törvényszerűségekre alapozva, adott tervezési szituációkra vonatkozó konkrét megállapításokat tud tenni.

Az emberi kreativitás számítógépes megfelelőjének létrehozása feltételezi a kreativitási folyamat megértését és modellezését. A feladat lényegében a kreativitást támogató rendszerek, pontosabban környezetek fejlesztésében jelölhető meg. E kreatív környezetben az ember és a munkáját támogató számítógépes rendszer egyaránt részt vesz az alkotási folyamatban. Általánosan elfogadott, hogy az alkotás transzdiszciplináris együttműködést igényel, amelynek a kreativitást támogató rendszerekre is ki kell terjednie.

Taníthatóság és öntanulás

A tanulás kapcsán viszont Carbonell és munkatársai az alkalmazott stratégia alapján megkülönböztették a taníthatóságot és az öntanulási képességet. A fejlesztők általi átprogramozás vagy új

tudáselemekkel kiegészítés a tanulás legalacsonyabb szintjét képviseli.

Lényegében olyan, mint az ember esetében a szorzótábla bemagolása (ezért is illeti az angol ezt a szintet a *rote learning* megjelöléssel. A másik véglet a teljesen felügyelet nélküli tanulás lenne (az *unsupervised learning*), ami a tudományos felfedezésekig is elvezethetne.

Amit a jelenleg ismert mesterséges intelligenciás tanulórendszerek meg tudnak valósítani, az inkább a taníthatóság valamilyen formája, semmint az öntanulásé. Vagyis messze áll attól a spontán folyamattól, amelyre az ember képes. Az öntanítás területén lényeges eredményt — ismereteim szerint — csak a versengő neurális hálókkal értek el. A tanulás megvalósításához a tanulási folyamatot is modellezni kell. Mint említettük, a tanulási folyamat szervesen beépül a tervezés folyamatába, ebből kifolyólag a tanulás modelljét integrálni kell a tervezési folyamat modelljébe. A beépíthetőségnek értelemszerűen előfeltétele, hogy a meglévő tudásra alapozott következtetés és az új tudás megszerzésére irányuló tanulás mechanizmusa összeférő legyen. Mindemellett izgalmas kérdés az is, hogy miként valósítható meg a tervezői praxis komplexitását kezelni tudó számítógépes tanulás.

Az egyik lehetőség az indukció

A jelenségek túlegyszerűsítése miatt hosszú ideig feltételezték azt, hogy a tervezési folyamat a formális logika következtetési folyamataival kimerítően helyettesíthető. Emlékeztetni szeretnénk arra, hogy a szakértőrendszerekkel foglalkozó írásban (lásd a májusi szám 51. oldalán) a logikai következtetés meghatározó jelentőségű következtető sémájának a — lehetőségei tekintetében lényegében feltárt — dedukciót állítottuk be.

A dedukció olyan formális logikai művelet, amely adott tények (a feltételek) összefüggései alapján más tényre (a következményre) utal. A dedukció teljesértékűsége abból a szempontból kifogásolható, hogy mind a feltételek, mind a lehetséges következmények csak ismert tények alkotta halmazból származtathatók. Vagyis hipotetikus tudás, ami érezhetően a kreativitás kulcsa, nem jelenik meg.

Az indukció az a következtetési fajta, amely közelebb visz bennünket a tanuláshoz — abból kifolyólag, hogy adott tényekből vagy példányokból általános elveket vagy koncepciókat alakít ki. Az induktív tanulást a szimbolikus leírások halmazán végrehajtott heurisztikus keresés jellemzi, amit különféle következtetési szabályoknak az eredeti leíró állításokra való alkalmazásával valósítanak meg. Ily módon mintegy félúton helyezkedik el a taníthatóság és az öntanulás között.

Az induktív tanulás

Így tanulni példákon keresztül lehet. A tanulási feladat célját a megfelelő koncepció felismerése vagy megszerkesztése jelenti. A példák megerősítő és tagadó jellegűek egyaránt lehetnek, sőt a kettő együttes alkalmazása hatékonyabban behatárolja a koncepciót. Az induktív tanulás során a példákban alapvetően két sajátosságegyüttest kell felismerni, mégpedig a strukturális és a szubsztanciális sajátosságokat. A strukturális sajátosságok a példa részleteit képező alapegységeket és a közöttük lévő (szerkezeti) kapcsolatokat fedik le.

A tanulórendszer tervezőjének kell meghatároznia, hogy egy adott osztályhoz tartozó példák (konstrukciók) esetén milyen alapegységeket azonosít, és rajtuk milyen topológiát értelmez. A szubsztanciális sajátosságok az objektumok tudományos eszközökkel felismerhető tulajdonságait fedik le. Ezek keretekbe foglalása, szabályokkal való megfogalmazása, vagy egyéb szimbolikus megragadása lehetővé teszi a példák formális kezelését.

Az induktív tanulással megszerzett tudás a konstrukció tervezésében a megoldási részletek megtalálása, és nem azok eredeti létrehozása szempontjából hasznosítható. A konstrukciós tervezésben általában a formalizáltan megfogalmazott működési igényekből indulunk ki, amelyeket a szaknyelv funkcionális követelményeknek nevez. Minden konstrukció adott funkcionális követelményegyüttes kielégítésére képes, és ez behatárolja a szerkezeti struktúráját és a tulajdonságait is. Mivel a konstrukcióra vonatkozó követelmények a szemantikai tartalmukból következő összefüggések megtartása mellett diszkretizálhatók, a követelményeket kielégítő szerkezeti részekre vonatkozó ismeretek megfelelő példák alapján elsajátíthatók.

Természetesen itt egy rendkívül hosszadalmas és elképzelésigényes folyamatról beszélünk. Ha a rendszert a követelmények és a megoldási részletek összefüggésére a példákon keresztül sikerült rávezetni, akkor az adott követelménycsoportokra megoldásokat javasolhat. A tanítás előkészítéséhez a példákat karakterisztikus részleteikre kell bontani, és ezeket a részleteket a vonatkozó követelményeknek meg kell feleltetni.

Le- és előíró általánosítás

Elvi szempontból az induktív tanulásnak a tervezésben két fajtája alkalmazható. Ezeket leíró (deszkriptív) általánosításnak és előíró (preszkriptív) általánosításnak nevezzük. E két forma eltérő tervezési problémafelvetésnek felel meg.

A deszkriptív általánosításra alapozott tanulás egy adott osztályhoz tartozó objektumok tulajdonságainak megállapítására irányul. Szaknyelven ezt az egyedek tulajdonságai elsajátításának nevezik, aminek alapja az osztály formális definíciójának létezése. A preszkriptív tanulási forma a tulajdonságok alapján objektumok osztályhoz tartozásának meghatározására irányul. Ezt másképpen az osztálykoncepció elsajátítása problémájának nevezzük. A végrehajtás alapja a változatterek definiálása. Az induktív tanulás újszerű logikai kalkulust igényel. Ezt változó-értékű logikának nevezzük, amelynek gondolata és eszközei először R. S. Michalski munkáiban jelentek meg még a hetvenes évek közepén. E logikai kalkulus alapeleme a *szelektor*, ami változókat (attribútumokat) és referencialistákat rendel össze relációs operátorokon keresztül.

A példákra alapozott induktív tanulást módszertani szempontból a korábbiakban már véleményeztük. A teljesség kedvéért meg kell említenünk egy hozzá kapcsolódó elvi korlátozást is. Nevezetesen arról van szó, hogy a példákkal való tanulással az adott követelményeket teljesítő funkcióhordozónak egyidejűleg csak egyetlen formalizált megvalósulási alternatívája sajátítható el. Így viszont a tervezési szintézis során például egy mechanikus működési elvű konstrukció alternatívájaként sohasem merülne fel egy hidraulikus megoldású. Ez viszont a mérnöki praktikum szempontjából általában erős megszorításnak ítélhető.

A másik lehetőség, az abdukció

Felvetődik a kérdés: ténylegesen a tervezési tevékenységegyüttesnek van-e meghatározó szerepe a tervezésben, vagy létezik valami más, aminek ugyancsak megvannak a belső törvényszerűségei, csak éppen a tevékenységre való összpontosítás miatt árnyékban marad? A kérdés már utal a válaszra.

Igen, a mesterségesen létrehozandó objektum tulajdonságainak mibenléte és együttese erősebben meghatározó, mint a hozzá kapcsolódó tevékenység — bármennyire is szokatlannak tűnik ez elsőre.

A dolog hasonlít ahhoz, ahogy a szobor benne van a kőtömbben vagy a farönkben. Az elképzelt lényeg határolja be a létrehozási tevékenységet. Hogy ezt a szemléletet megvalósíthassuk, új paradigmát kell alkalmaznunk. Az alábbiakban erre egyszerűen úgy hivatkozunk, hogy abduktív módon következtetünk. Az abduktív következtetés azonban az elsajátítás ismertetőjegyeit is magán hordozza.

Az abdukció gondolatát C. Peirce a harmincas évek közepén fogalmazta meg. Értelmezése szerint az abdukció esetében a következtetés iránya a dedukcióéval ellentétes. Tudásfeldolgozó rendszerbeli megvalósításának kérdéseivel H. Pople a hetvenes évek elején született cikkében foglalkozott. E két kutató gondolatai közvetlenül hasznosíthatókká válnak a tervezésben, ha elfogadjuk, hogy annak lényegét a szintézis adja.

A szintézist erősebben befolyásolja az elképzelt tárgya, mint a hozzá kapcsolódó tevékenység. Az abdukció elengedhetetlen ahhoz, hogy tényleges tervezési szintézist tudjunk megvalósítani. Az abdukció lényegét az objektum tulajdonságaira és viselkedésére vonatkozó tudás megragadása és feldolgozása képezi. Míg a dedukció ismert megoldások megtalálását teszi lehetővé, az abdukcióval nem ismert megoldások is kikövetkeztethetők. Az abdukció az objektum leírásait hipotézisként állítja elő. Az elvi háttér sajnos csak erős absztrakcióval tárgyalható.

Objektum és...

A kreativitás abduktív következtetéssel való támogatására vonatkozó elképzelést a The University of Tokyo, Research into Artifacts, Center for Engineering (RACE) munkatársai alakították ki. Megközelítésük középpontjában az objektumorientáltság áll. Az objektum fizikailag létező dolog, míg az objektumkoncepció az embernek az objektumra vonatkozó elvonatkoztatott mentális elképzelése. Az objektum tulajdonságai az absztrakt koncepciók, amelyeket attribútumokkal lehet kifejezni.

Az abduktív következtetés megvalósításához szükség van az objektumkoncepciók adatbankjára. Ennek az adatbanknak az objektumkoncepciók közötti kapcsolatokat is tükröznie kell. Ez utóbbi eredményezi az ember asszociatív agymodelljére emlékeztető kapcsolati gráfot. A tervezési folyamat lényegét ebben a megközelítésben az adja, hogy a funkciókoncepciók halmazán értelmezett topológiák alapján, az absztrakt koncepciók segítségével új objektumkoncepció-topológiákat hozzunk létre.

Háromrétegű neurális háló

A megvalósításban a japán kutatók a neurális hálók elvét adaptálták. Az abduktív tervezőgép tehát lényegében egy speciális kialakítású neurális háló. Olyan háromrétegű hálót szerkesztettek meg, amelynek egyik rétege a szülő, a másik a nevelő, a harmadik a partner réteg. A tervezett objektum létrehozása e három rétegen való átnyomulással érhető el.

A szülő réteg tartalmazza az objektumkoncepciókat (azaz megfelel azok adatbankjának), és megadja a tervezésben felhasználható objektumok közötti szintaktikus kapcsolatokat is. Tulajdonképpen a második rétegen valósul meg az abdukció. E nevelő réteg az objektumkoncepcióra jellemző attribútumokat tartalmazza. A harmadik réteg csomópontjai a funkciókra vonatkozó formális megfogalmazásokat tartalmazzák. A háló ezen a partner rétegen tart kapcsolatot a külvilággal. Ha (a követelmények formális megfogalmazása alapján) a bemeneti rétegen megjelenik a létrehozott funkciómező, a közbenső réteg csomópontjaiba (azaz a tulajdonságegységekhez) továbbított aktiválás hatására az attribútumoknak a funkciókoncepciók beérkező topológiájának megfelelő összerendeződése alakul ki. Attól függően, hogy milyen volt a funkciók topológiája, más és más attribútum-összerendeződés alakul ki. A szülő rétegen — a továbbított aktiválás hatására — az objektumkoncepciók összerendezése ennek megfelelően történik meg. A feldolgozás eredményét az objektumkoncepciók új topológiája adja.

Az abduktív következtetés a neurális hálók előretáplálással való taníthatóságának elvét hasznosítja. Itt kell megemlíteni, hogy az abduktív következtetés neurális hálóra alapozott megközelítése feloldja az induktív tanulórendszer unicitását (egymegoldás-orientáltságát). A rétegek közötti átvitelt súlyértékekkel lehet befolyásolni. Ha annak szubjektív valószínűsége, hogy két aktiválás egyenlő fontosságú, sokkal nagyobb, mint annak, hogy különböző, az aktiválás súlyértéke nagy pozitív érték. Ellenkező esetben nagy negatív érték. Ha a két koncepció független, az aktiválás súlyértéke zérus.

A súlyok kezelése alapján látható, hogy az abduktív tanulás a versenyeztetés elvét alkalmazza az aktiválásban. A (megoldási) nevelő rétegen kialakult attribútummező a rétegek csomópontjai közötti kapcsolatokon keresztül jut át a szülő rétegre. Mivel ezen a fizikai megvalósítás szempontjából elfogadható objektumkoncepciók találhatók, e réteg lényegében végrehajtja a megoldás konszolidációját is.

Az eddigiek alapján már kimondható: az abdukció azért újszerű és fontos, mert a tulajdonságokat állítja elő, és nem azt, hogy miképpen kell tervezni. Az alkalmazásokba is beépült eredményekre azonban még várnunk kell.

Horváth Imre

A MTESZ „behálózása"

A Műszaki és Természettudományi Egyesületek Szövetségi Kamarája (MTESZ) korszerű számítástechnikai eszközökből álló hálózatot hozott létre a MTESZ tagegyesületei és általában a műszaki értelmiség számára. Ez hozzáférést biztosít a külföldi információs szolgáltatások széles köréhez, korszerűsíti a belső levelezést és az irodai funkciókat.

Jelenleg tíz vidéki Technika Házában (Eger, Esztergom, Győr, Kecskemét, Miskolc, Pécs, Szeged, Székesfehérvár, Szolnok és Vác), valamint a budapesti központban helyeztek üzembe ún. információs pultokat (közepes teljesítményű, Intel alapú desktop számítógép, CD-ROM olvasó, lézernyomtató, faxmodemkártya, telefonfővonal, alap- és alkalmazási szoftverek alkotta együttes).

Az Internetet a MTESZ ingyenesen használhatja. A központi Unix alapú gépet az információs pultok modemen keresztül érhetik el. A Unix gép egy útválasztón (router) keresztül csatlakozik az IIF Internet csomóponti számítógépéhez, és azon keresztül a teljes Internet hálózat elérhető.

A további tervekben szerepel valamennyi Technika Házának bekapcsolása a rendszerbe, továbbá a MTESZ tagszervezetei részére — azok saját eszközparkjával — az Internet csatlakozás felajánlása. A legaktuálisabb információkat TeleDataCast (TDC) szolgáltatással juttatják el az információs pultokhoz.

A matematika már holt nyelv?

Legfontosabb fogalmaink

Érdemes lenne egy szakmai közvéleménykutatást végezni, hogy melyek szakmánk legfontosabb fogalmai, és ezeket ki hogyan értelmezi. Így fény derülne arra a kaotikus állapotra, amelynek megdöbbentő jeleivel naponta találkozhatunk, okaival azonban senkinek sem akaródzik foglalkozni.

A +, –, * és a / műveleti jeleket (vagy ha tetszik műveleteket) relációkként emlegetik egyes tudós számítástechnikus szerzők. A <, ≤, >, ≥ relációjeleket (vagy ha tetszik relációkat) pedig más, nem kevésbé tudós számítástechnikus szerzők viszont a műveletek közé sorolják. Az első csoport számára minden reláció, a második számára pedig minden művelet... Egyik csoport sincs igazán tisztában szakmánk két legfontosabb szakkifejezésének jelentésével, ez azonban nem gátolja őket a tankönyvírásban.

A szakmai műveletlenség ilyen és hasonló példáival köteteket tölthetnénk meg — biztosan lesznek majd, akik elvégzik ezt a tudománytörténeti és egyben kortörténeti szempontból hasznos munkát. Most azonban fontosabb lenne a rendcsinálás. Legalább azt el kellene érnünk, hogy szakmánk legfontosabb szakkifejezéseit mindenki helyesen használja. Nem lenne felesleges egy néhány oldalas miniszótár összeállítása sem. E szótárban mindenekelőtt azok a szavak (fogalmak) kaphatnának helyet, amelyekkel szakmánk legfontosabb és leggyakoribb történéseit, jelenségeit megragadjuk, leírjuk (amelyekkel a műveletleírást, helyzetleírást, viszonyleírást végezzük, amelyek a legegyszerűbb nyelvi műveletmodellnek, illetve szituációmodellnek foghatók fel stb.). Ennek a szótárnak az összeállításához kívánunk hozzájárulni a következő, legfontosabb számítástechnikai fogalmainkkal kapcsolatos értelmezési észrevételekkel.

Művelet

Idegen szóval operáció; három lényeges összetevője van: (1) az operátor, amely a műveletvégzést (le)bonyolítja, (2) az operandusok, amelyekkel a műveletvégzés történik, és (3) az operátum, illetve operátumok, amelyek a műveletvégzés eredményei.

Reláció.

Magyar megfelelői a kapcsolat, összefüggés, viszony, vonatkozás; nyelvi megjelenési formája a kijelentés, az állító mondat; helyzet-, illetve szituációleírásra használjuk, helyzet-, illetve szituációleírási eszköz; amikre a reláció vonatkozik, relátáknak hívjuk, amit a relátákról kijelentünk, állítunk, az a reláció (nem szerencsés módon a helyzet-, illetve szituációleírási eszköznek és az egyik alkotórésznek ugyanaz a neve, célszerűbb lenne az utóbbit relációállítmánynak hívni).

Hozzárendelés

Művelet, melynek eredményeként hozzárendeltségi reláció jön (jöhet) létre.

Függés

Rendszerjellemzők közötti kapcsolatban megnyilatkozó jelenség; a függés lehet időtől függő, múltbeli történésektől függő, és a legegyszerűbb esetben (Dirichlet-féle) matematikai függvénykapcsolattal leírható.

Függvény

Többjelentésű kifejezés, jelenthet egy viszonyt (egy ún. függvénykapcsolatot is, és e viszony egyik relátáját is. Egy mai definíció: Adott két halmaz, H és K. Ha a H halmaz minden egyes eleméhez hozzárendeljük a K halmaznak egy-egy elemét, akkor a hozzárendelést függvénynek nevezzük. Ez kétszeresen is hibás. Egyrészt függvény helyett függvénykapcsolatot kellene mondani, másrészt pedig a hozzárendelés művelet, holott a függvénykapcsolat reláció. Ne felejtsük el Dirichlet eredeti (1837-ből származó), soha érvényét nem vesztett definícióját: azt mondjuk, hogy y az x egyértékű függvénye, ha x minden szóba jövő értékéhez y-nak meghatározott értéke tartozik. Ez tehát egy helyzet: valamihez valami, valamikhez valamik tartoznak. E helyzetben semmi szerepe nincs annak, hogy az hogyan jött létre. Egy állapotra vonatkozó puszta ténymegállapítással van dolgunk. Állapotot pedig relációkkal, nem pedig műveletekkel írunk le. Egy-egy operandus és operátum is egymáshoz rendelhető, ezáltal függvénykapcsolat definiálható, ily módon egy operációval reláció generálható; az operáció e reláció generáló operációja lesz, de természetesen nem válik ezzel relációvá, tehát függvénykapcsolattá. Sem a függvény(kapcsolat) nem operáció, sem az operáció nem függvény(kapcsolat), annak ellenére sem, hogy nemcsak operációval generálható függvénykapcsolat, hanem függvénykapcsolattal is generálható operáció.

Változó

Hibás fordítás eredményeként elterjedt kifejezés; helyesebb lenne „változtatható”-t mondani; az angolban is a „variable” nem változót, hanem változhatót jelent. Relációk relátáit szokták változóknak nevezni. Szerepe van még a függvénykapcsolatban is. Kétféle szóhasználat van forgalomban (egyik sem tökéletes). Az egyik szerint a függvénykapcsolatban az, amihez tartozik valami, a „független változó”, ami pedig ehhez tartozik, az a „függő változó”. A másik szóhasználat a „független változó” helyett „változót” mond, a „függő változó” helyett „függvényt”.

Transzformáció

Magyarul átalakítás, mint ilyen, operáció. Gyakran összetévesztik az operáció operandusa(i) és operátuma(i) közt definiálható relációval, függvénykapcsolattal. Különösen visszatetsző ez a geometriai transzformációk esetében, amikor például azt mondják, hogy „eltolnak egy pontot” pedig valójában az történik, hogy „tükröznek egy pontot”, ugyanis a pont megmarad ott, ahol volt, csak megszerkesztik a tükörképét is. Hasonlóan ahhoz, amikor egy számítástechnikus arról beszél, hogy „átvisz

egy adatot egyik regiszterből egy másikba", pedig közben az átvitt adat ott is megmarad, ahol volt — vagyis helyesebb lenne ilyen esetekben az adat átmásolásáról beszélni.

Automata

Műveletvégző rendszereknek, illetve rendszerek viselkedésének leírására szolgáló fogalom; lényegében operátor.

Halmaz

Alapfogalom; meghatározott, különböző dolgok összességét jelenti. Korunk egész matematikáját megpróbálták erre a fogalomra felépíteni. A gyakorlatban azonban csak nagyon korlátozottan alkalmazható. Már a természetes számok tízes számrendszerbeli felírásával is problémák adódnak. Honnan vegyük pl. azt a 3 darab 1-es számot, ami az 121451 szám felírásához kell? Tudjuk, a számok számjegyei is a természetes számok halmazának elemei. Abban viszont, mivel az halmaz, minden elem csak egy példányban fordulhat elő, tehát halmazelméleti alapon még egy valamirevaló számot sem tudnánk felírni. Egyetlen kiút kínálkozik, hogy a természetes számok halmaza is több példányban van meg. Hány példányban? Végtelen sokban? Hogyan lehet ezeket a halmazokat megkülönböztetni? Ha nem lehet megkülönböztetni, honnan tudjuk, hogy hány példányban vannak meg?

A szakkifejezések után érdemes még néhány szót szólni szemléletmódunkról is. Arról, hogy mikre figyelünk, mikben és hogyan gondolkodunk. Mindeddig nagyon is a matematika bűvöletében éltünk, annak ellenére, hogy a matematika egzaktság iránti igénye sosem nyert túl nagy teret a számítástechnikában. Jobbára számokban gondolkodunk, ha ez nem elég, akkor vektorokban, ha még ez sem, akkor mátrixokban.

Célszerűbb ennél egy gyakorlatiasabb gondolkodás: amiről szó van, azt a benne levő relációkkal és azokkal a relációkkal együtt lássuk, amelyekben ennek a valaminek szerepe van. Így nem számok, vektorok, stb. állnak majd a munka középpontjában, hanem természetes rendszerek, mezőnyök, eloszlások, elosztások, részesedések, összetételek, elhelyezések, elhelyezkedések, ezek jellemzői és a felsoroltak kalkulusai.

Ez annál is természetesebb, mert hiszen állandóan küzdelmet folytatunk. És nyilvánvaló, hogy bármilyen küzdelemben a küzdő felek nem tehetnek mást, mint a rendelkezésükre álló erőforrásokat, eszközöket használják, elosztva azokat különféle feladatok (részfeladatok) megoldására. Tehát a küzdelem lényege sem más, mint erőforráselosztás (és felhasználás). A küzdelem sikere is lényegében ettől az erőforráselosztástól (és felhasználástól) függ. (Lényeges tudatosítanunk, hogy minden küzdelem célja is mindig elosztás. Minden küzdő fél arra törekszik, hogy a javak és terhek elosztása az ő számára az elérhető legelőnyösebb legyen.)

A számítástechnika gyakorlati tudomány. Gondolkodásának is gyakorlatinak kell lennie. Egészen konkrétan: a számítástechnikai gondolkodásnak, kommunikációnak gyakorlati szempontok szerint „strukturáltnak" kell lennie. Szavakkal is kifejezetten kell jeleznünk, hogy amiről éppen szó van, az például helyzet, lehetőség, eszköz, igény, probléma, feladat, viselkedés, működés (navigáció, gazdálkodás), approximáció, eljárás (ne csak matematikai algoritmusokról jusson eszünkbe ez a szó!), jellemző, jellemzés, igénykielégítés, operandus, operátor, operátum, reláció, reláta stb. A gondolkodásmód megváltozásához az oktatási szemléletnek is meg kell változnia: nem matematikát és számítástechnikát, hanem matematikahasznosítást és számítástechnika-hasznosítást kell tanítani.

Kritikus sorok után illő dolog enyhítő körülmények után nézni. Van ilyen is, ez a vétőket menti valamelyest, de a következményeket nem enyhíti. Mindannyiunk fő mentsége, hogy gyakran beleesünk a nyelv csapdáiba.

Leonardo da Vincitől származik az a kíméletlenül kemény megállapítás, hogy „aki a matematika segítségét megveti, az zavarosban halászik...". Ez ma már nem igaz, mert ma már a matematika régen nem az, mint egykor volt, a „természettudományok nyelve". Korunk matematikája máris félig holt nyelv a természettudományok számára.

Képes lesz-e a számítástechnika a legfontosabb és leggyakoribb történések, jelenségek stb. kielégítő egzaktságú leírására? Ettől függ majd, hogy sikerül-e, és mikorra sikerül kilábalnia a jelenlegi kaotikus állapotból, amikor még az alapfogalmak terén sincs rend. Ha sikerül a számítástechnikának a matematikai egzaktságot elérnie, átveheti a matematika szerepét. Nem úgy, hogy a számítástechnika lesz a természettudományok nyelve, hanem úgy, hogy a számítástechnika lesz az egzakt gondolkodás terjesztője. Kár lenne ezt a történelmi lehetőséget elszalasztanunk.

Pogány Csaba

A Fortran 90 nyelv — II.

Mutatós változók

Az algoritmikus nyelvek közül elsőként a Fortran vált világszabvánnyá, ezen státust kiérdemlő változat dátumára utaló kiegészített névvel. Ez a tény, és az, hogy megszületett a PC-re írt legelső fordítóprogram is, olyan lehetőséget hordoz a transzportábilis termékek készítését szorgalmazók, illetve ilyen programokra „ácsingózó" felhasználók szempontjából, hogy a téma külön figyelmet érdemel.
A múlt hónapban elkezdett sorozat második része most ezzel foglalkozik.
Szeptemberben, a sorozat harmadik (egyben befejező) részében igyekszünk majd tömören elmondani a legfontosabb tudnivalókat e termékről

Emlékeztetjük olvasóinkat, hogy az ismertetésben a Lőcs-Vigassy könyv alapján haladunk, ahol lehetséges. Ezért például az alábbiakban, amikor az utasításokat tárgyaljuk, el kell térnünk a Fortran 90 szabványos utasításcsoportjaitól, hogy kövessük a tankönyv logikáját. A *végrehajtható utasítások* közül az alprogramkezelő és I/O utasítások később kerülnek sorra.

Értékadó utasítások (o e)

Az értékadó utasítás végrehajtásaként először kiértékelődik a jobb oldali kifejezés, majd az = egyenlőségjel bal oldalán álló változó megkapja a kifejezés értékét, például i=2*3.

Numerikus változó csak numerikus értéket kaphat. Ha a kifejezés fajtája más, mint a változóé, implicit típuskonverzióra kerül sor. Karakterváltozó csak ugyanolyan fajtájú karakterértéket kaphat (például egy ASCII-fajtájú változó nem kaphat cirill betűs szöveget). Eltérő hosszúság esetén a karaktersor vége elvész, illetve szóközökkel töltődik fel. Logikai változó csak logikai értéket, a felhasználó által definiált változó csak ugyanolyan típusú értéket kaphat.

Skalárok összeillenek minden tömbbel, vagyis az A = 1.23 utasítás hatására az A tömb minden eleme megkapja az 1.23 értéket. (Fordítva természetesen nem működik; ha x skalár, az x = A hiba.)

Ha a változó a Fortran 90-ben bevezetett mutató (pointer), a célváltozó (target) kapja meg a kifejezés értékét. Ehhez természetesen előzőleg össze kell őket rendelni (lásd alább).

Vezérlőutasítások és címkék (❑ d)

A címkék változatlanul 1–5 jegyű egész számok. Ezekre át lehet adni a vezérlést. Újdonság, hogy bizonyos programbeli szerkezeteknek nevet lehet adni (ezek formája ugyanaz, mint a változóké), a szerkezetet azonosító Fortran 90-beli alapszótól kettősponttal elválasztva. Ez a név csak a szerkezettel kapcsolatban van értelmezve, például a ciklusutasítások hatáskörével kapcsolatos összefüggésekben.

Az *assign* utasítás a programok transzportja érdekében működik, de elavult: új programokban kerülendő. Hasonló a helyzet az aritmetikai *if* utasítással is. A logikai *if* és a kiszámított *go to* utasítás szerepe, noha manapság minden gép egycímes, nem változott.

A Fortran 77-ben bevezetett blokk-if utasítás része a Fortran 90-nek is. Az utasítás lehetővé teszi a végrehajtandó utasításcsoport (blokk) egy vagy több logikai feltétel szerinti kiválasztását. Az utasítás formája:

```
[ szerkezet-név : ] IF (skalár logikai kifejezés) THEN
 blokk
[ ELSE IF (skalár logikai kifejezés) THEN
[szerkezet-név ]
 blokk ]
...
[ ELSE [ szerkezet-név ]
 blokk ]
END IF [ szerkezet-név ]
```

ahol a „szerkezet-név" a Fortran 90-ben „IF-szerkezet"-nek nevezett programrészlet neve, a „blokk" pedig vagy üres, vagy egy vagy több végrehajtható „szerkezetet" tartalmaz.

Hasonló a feladata a Fortran 90 esetkiválasztó szerkezetének (Case construct). Ez egy skaláris integer, logical vagy character kifejezés értékét vizsgálja, és attól függően, hogy az a feltételekben megadott melyik tartományba esik, kiválasztja a végrehajtandó blokkot. Formája (❑ d,j):

```
[ szerkezet-név : ] SELECT CASE ( kifejezés )
CASE ( eset-választó [, eset-választó ]... ) [
szerkezet-név ]
 blokk
...
[ CASE DEFAULT [ szerkezet-név ]
 blokk ]
END SELECT [ szerkezet-név ]
```

Az „eset-választó" megfelelő típusú skaláris konstans vagy tartománykijelölés. A tartományokat legegyszerűbb egy példával illusztrálni:

```
select case (i)
case (:-2) ! i <= -2
...
case (0) ! i = 0
...
case (4:8) ! 4 <= i <= 8
...
case default ! i = -1, 1, 2, 3, 9, 10...
...
end select
```

A Case defaulthoz tartozó utasítások akkor kerülnek végrehajtásra, ha a többi eset közül egy sem teljesül.

Rokona ezeknek az új feltételes tömbértékadó utasítás (masked array assignment, ❑ d):

```
WHERE ( tömbre vonatkozó logikai kifejezés)
 [ értékadó utasítás ] ...
ELSEWHERE
 [ értékadó utasítás ] ...
 END WHERE
```

Az egyszerűbb forma a logikai IF-re emlékeztet:

```
WHERE ( tömbre vonatkozó logikai kifejezés ) értékadó
utasítás
```

Példák (a, b és c tömböket jelölnek):

```
where (b<=c) ; b=2 ; elsewhere ; c=-3.45 ; end where
   where (a>b) a=-1
```

Indexes változók, tömbök, indexkifejezések

A Fortran 66 indexes változóinak jelölése változatlan. Az indexek nélküli tömbnév már akkor is az egész tömböt jelentette. Újdonság, hogy a két véglet, az egyetlen tömbelem és az egész tömb között igen egyszerű írásmóddal, szinte korlátozás nélkül definiálhatunk tömbszeleteket (array section), például egy négydimenziós tömb egyik indexének rögzítésével nyert háromdimenziós tömböt.

Annak mintájára, ahogyan egy skalár-értékadásnál felírhatunk a jobb oldalon egy számot, amelynek nincs neve, használhatunk névtelen tömböket is (array constructor, it is an unnamed array). Formájuk:

```
(/ tömbbé szerkesztendő értékek listája /)
```

a lista pedig kifejezésekből és belső ciklusokból áll. Példák: lista: (/1,2,3,4/), belső ciklus: (i,i=1,99,2), 2*2-es mátrix létrehozása 4 adatból (❑ i): c=reshape((/4,2,3,1/),(/2,2/))

A Fortran 66 által „elismert" indexkifejezések az elemi index (element-subscript) kategóriának igen kis részhalmazát alkotják, hiszen tetszőleges skalár integer kifejezés volt használható már a Fortran 77-ben is. Az utóbbihoz képest is minőségi ugrást jelent az indexhármas (subscript triplet) és a vektorindex (vector subscript) bevezetése a tömbszeletek kijelölésére. Az indexhármas formája:

```
[ elemi index ] : [ elemi index ] [ : lépés ]
```

ahol a lépés is tetszőleges skalár integer kifejezés. A három adat értelmezése megfelel a *do* utasítás három adatának; ha az első hiányzik, az alsó indexkorlát helyettesíti, a másodikat a felső korlát, a harmadikat pedig 1. Példák:

```
a(2:5)  ! a(2), a(3), a(4), a(5)
b(11:3:-4)  ! b(11), b(7), b(3)
c(1:3:2,:) ! c első és harmadik sorából képzett mátrix
```

Még ennyi szabályosságot sem követel meg a vektorindex használata: az egydimenziós tömbben (vektorban) felsorolt indexek jelölik ki a tömbszeletet, például: vektor=(/1,4,2,5/);c=b(vektor,3) ! b(1,3), b(4,3), b(2,3), b(5,3)

A jobb memóriagazdálkodást támogatja az *allocate — deallocate* utasításpár (❑ e,j). Ezek használatával (az ilyennek deklarált) tömböknek futásidőben foglalhatunk le memóriát, szabadíthatjuk fel a lefoglaltat.

Ciklusutasítások és használatuk

A Fortran 66 *do* utasítása formálisan változatlan, de működése a Fortran 77-ben előírtaknak felel meg. A ciklusváltozó értékének és végértékének összehasonlítása a ciklusmag végrehajtása előtt történik meg. Ez azt jelenti, hogy az i=2,1 és i=1,2,–1 értékekkel a ciklus egyszer sem fut le. Adott esetben a *do* utasítást át kell írni a 66-os stílust szimuláló elemi utasításokká. (Vigyázat: a Fortran 66 a ciklusmag után tesztel! A ciklusszerkezet végére kell írni az „IF (ciklusváltozó < felső határ) THEN ciklusváltozó = ciklusváltozó+lépés; GO TO eleje ; END IF" programrészletet.)

A Fortran 90 bővítései a ciklusutasítás nevében is megjelennek: DO szerkezetként (DO construct) hivatkoznak rájuk (❑ d). Általános formájuk:

```
[ szerkezet-név : ] DO [ címke ] [ ciklusvezérlés ]
 blokk
{ END DO [ szerkezet-név ] | címke záróutasítás }
```

A ciklusvezérlés:

```
{ [,] ciklusváltozó = kifejezés,kifejezés [,kifejezés]
|
[,] WHILE ( feltétel ) }
```

A ciklusvezérlés első formája a régi. A második formában szereplő feltétel skaláris logikai kifejezés. (A Fortran 77 által megengedett REAL ciklusváltozó használható ugyan, de elavult, kerülendő. A processzorfüggő kerekítési hibák miatt a program másik gépen másképp viselkedne.) A ciklusmag mindaddig végrehajtódik, amíg a feltétel értéke .TRUE. A záróutasítás végrehajtható utasítás (tipikusan *continue*), de nem lehet *cycle, end, exit, goto,* kijelölt *go to*, aritmetikai *if*, *return* és *stop* utasítás. Ha a ciklusvezérlés hiányzik, a ciklus végtelen (a befejezést a ciklusmagban kell programozni).

A címkézetlen cikluszáró utasításra (end do) lehet a ciklusmagból kiugrani a cycle [szerkezet-név] utasítással. Ezzel szemben az exit [szerkezet-név] utasítás befejezi a ciklusutasítást.

```
i_ciklus : do i=1,100
  kozbenso : do while (ab)
    do 33 j=1,100,3
      ...
      if (valami) cycle kozbenso
      ...
    33 continue
  end do
end do i_ciklus
```

A Fortran 66 „kiterjesztett hatáskörű ciklusát" már a Fortran 77 megtiltotta, így természetesen a Fortran 90 sem ismeri. Szükség esetén a programot — a 66-os stílusú ciklus szimulálásának mintájára — úgy kell átírni, hogy ne legyen benne *do* utasítás.

Megállító utasítások (pause és stop)

A *pause* [{ üzenet | legfeljebb 5 jegyű egész szám }] utasítás hatására a program futása megáll, majd operátori beavatkozásra folytatódik (PC-k esetében ez az Enter leütése). Mivel a PC-k kivételével minden számítógép multiprogramozású, az utasítás elavult, kerülendő.

A *stop* [{ üzenet | legfeljebb 5 jegyű egész szám }] utasítás hatására a program továbbindíthatatlanul megáll. Megjegyzés: sok Fortran 77 fordítóprogram ezt az utasítást egyben a DOS errorlevel változójának értékadására is használja. A Lahey Fortran 90 fordítója esetében az üzenet, illetve szám csak a képernyőn jelenik meg, hiszen a nyelvi szabvány nincs a DOS-hoz kötve. Az errorlevel beállítása a fordítóprogramhoz tartozó könyvtár szubrutinjának hívásával történik:

```
call exit(13) ! set errorlevel=13.
```

A mutatók céljának kijelölése

A Fortran 90-nel bevezetett mutató típusú változók használata a cél kijelölése vagy új cél kijelölése után ugyanolyan, mint maguknak a korábbi Fortran-változatokból ismert vál-

tozóknak a használata. Ez azt jelenti, hogy a mutatókat ugyanúgy kell deklarálni, mint bármilyen másik változót, de a pointer tulajdonságot (❑ g) fel kell tüntetni (részletesen lásd a későbbiekben).

A mutatókijelölés (pointer assignment) operátora a „=>". A kijelölést hatálytalanítja a *nullify* utasítás (❑ e,j). Példa: a=>b ; c=>d ; ... folytatás ... ; nullify a,c.

A mutatók bevezetésével mentesülhetünk igen sok alprogramhívástól és/vagy blokk-if utasítás felírásától, továbbá a program módosítása is egyszerűbben oldható meg. Ha például exportterveink vannak, a „hét nyelven beszélő" programmal (❑ j,k) a nyolcadik nyelvet úgy taníthatjuk meg, hogy csak a deklarációs részt és a mutató kijelölését kell módosítanunk, például így: nyelv=>nyolcadik.

A Fortran 90 program szerkezete (❑ h)

A nem végrehajtható utasítások tárgyalását bevezetendő, néhány fontos dolgot tisztázunk.

Mint arról már szó esett, a Fortran nyelvcsalád legnagyobb előnye az, hogy lehetővé teszi a nagy programok részekre bontását, és a részek egyenkénti lefordítását, majd a lefordított programrészek összefűzésével létrehozott kód helyes működését. A Fortran 66 által bevezetett szerkezetek (főprogram, a Fortran 77-ben *program* szegmensnek elnevezve, továbbá *subroutine, function, block data* szegmensek) mellé a Fortran 90 definiálja a *module* szegmenseket is. A szegmensek hivatkozhatnak egymásra, vagy egymás részeire (private, public, use). Szerkezetüket a táblázat mutatja.

<table>
<tr><td colspan="3">PROGRAM, FUNCTION, SUBROUTINE,
MODULE, BLOCK DATA utasítás</td></tr>
<tr><td colspan="3">USE utasítások</td></tr>
<tr><td rowspan="4">FORMAT
és
ENTRY
utasítások</td><td colspan="2">IMPLICIT NONE</td></tr>
<tr><td>PARAMETER
utasítások</td><td>IMPLICIT
utasítások</td></tr>
<tr><td>PARAMETER
és DATA
utasítások</td><td>Leszármaztatott típusok definíciója
INTERFACE blokkok
Típusdeklarációs utasítások
Utasításfüggvények
Specifikációs utasítások</td></tr>
<tr><td>DATA utasítások</td><td>Végrehajtható utasítások</td></tr>
<tr><td colspan="3">CONTAINS utasítás</td></tr>
<tr><td colspan="3">Belső alprogramok vagy modul alprogramok</td></tr>
<tr><td colspan="3">END utasítás</td></tr>
</table>

A táblázat természetesen nem mutathat be minden szabályt, így például azt sem, hogy az alprogramok másodlagos belépési pontjait definiáló *entry* utasítás (❑ d) értelmetlen (és ezért tilos is) a *program* és *block data* szegmensekben.

Típus- és tömbdeklarációk (❑ g)

A Fortran 66 és Fortran 77 típus- és tömbdeklaráló utasításai (*integer, real, double precision, complex, logical és character,* illetve *dimension*) változatlanul használhatóak.

Mivel a program változóinak az említett utasításokkal leírhatóaknál sokkal több tulajdonsága lehet, a Fortran 90 egy teljesen új deklarálási formát is bevezetett. Ezzel egységesen lehet kezelni a nyelv belső típusait, és a felhasználó által definiált típusokat: { { INTEGER I REAL I COMPLEX } [fajtakiválasztás] I DOUBLE PRECISION I CHARACTER [karakterfajta-kiválasztás] I TYPE (típus-név) } [, tulajdonságok listája ::] akármi [, akármi] ...

A deklarált akármi valamilyen programbeli mennyiség. Mód van arra is, hogy akármihez kezdeti értéket rendeljünk. Az akármi:

```
{ akármi-neve [ ( tömb-specifikálás ) ] [ =
kezd-érték-kif ] |
 függvénynév [ ( tömb-specifikálás ) ] }
```

A kezdeti értéket meghatározó kezd-érték-kif olyan legyen, hogy fordítási időben ki lehessen számolni.

A fajtakiválasztás formája: (KIND = kifejezés). A dupla pontosságú REAL esetében természetesen nem szerepelhet, hiszen a DOUBLE PRECISION eleve a REAL egyik fajtája. A karakterfajta-kiválasztásban a hossz is szerepel, például így: (LEN = kifejezés , KIND = kifejezés). Ha a deklarációban nincs ilyen fajtaleíró rész, az alapértelmezés lép életbe.

A TYPE (típus-név) a felhasználó által előzőleg már definiált típushoz tartozó mennyiséget deklarál (lásd alább).

A tulajdonságok listája a következőkből válogatható össze:

— ALLOCATABLE [olyan tömbök, melyek helyfoglalása, ill. a lefoglalt hely felszabadítása később, a végrehajtható *allocate,* ill. *deallocate* utasítással történik, ❑ e,k],

— DIMENSION (tömb-specifikálás) [tömbdeklaráció, max. 7 index],

— { EXTERNAL I INTRINSIC } [eljárások],

— INTENT ({ IN I OUT I IN OUT }) [az eljárás paramétere bemenő, kimenő, be- és kimenő; a bemenőnek legyen értéke az eljárásba belépéskor, a kimenő értéket kap az eljárásban],

— OPTIONAL [az eljárás változó darabszámú argumentummal hívható, az itt felsoroltak elhagyhatók],

— PARAMETER [névvel hivatkozott konstans],

— { POINTER I TARGET } [mutató, ill. cél],

— { PRIVATE I PUBLIC } [a modulbeli változók kívülről hozzáférhetetlenek, ill. elérhetők],

— SAVE [a változó értéke megőrzendő az alprogramba való újra belépésig; SAVE nélkül, alapértelmezésben minden lokális változó értéke elvész].

A tulajdonságok értelemszerűen gyűjtendők egybe, hiszen például egy függvény neve nem lokális, vagyis SAVE-et tartalmazó listára nem vehető fel. A tulajdonságok külön utasításokkal is leírhatók.

A tömbspecifikálás módja ugyanaz, mint a jelentősen kiterjesztett tömbdeklarációs utasításban:

```
DIMENSION [ :: ] tömbnév ( tömb-specifikálás )
 [, tömbnév ( tömb-specifikálás ) ] ...
```

Tömbspecifikálás lehetséges explicit módon, az indexhatárok felírásával (ha nincs alsó határ, az 1-et jelenti); csak a dimenziószám megadásával (megfelelő számú : felsorolásával), ebben az esetben vagy a tényleges helyfoglalás az *allocate* utasításra történik, vagy az eljárás az aktuális paramétertömb adatait kapja meg; végül az eljárások paraméter-átvételeként megvalósuló dinamikus tömbdeklarációval. Példák:

```
dimension a(–1:1)
real, dimension (–1:1), save, target :: b=(/–1.1,0,1.1/)
real, dimension (–1:1), pointer :: c
real, dimension (:), allocatable :: d
subroutine e(f,g,n)
 real, intent(in):: f ; integer, optional :: g
 integer k,n ; dimension f(n),k(0:n*n)
```

Részletes példák találhatók a mintaprogramokban: ssp_minv_demo (❑ l,n), integral_ssp_romberg (❑ k,n), euler_demo (❑ l,n).

Az *equivalence* és a *data* utasítás

A Lőcs—Vigassy könyv érdekesen indokolja az *equivalence* utasítást: „lehetővé teszi, hogy a program végrehajtása során, időben egymás után a memóriában ugyanazon a címen több változót helyezhessen el, vagy hogy ugyanazt a mennyiséget egy időben több néven is nevezhesse." Az indoklás eleje a Fortran 90 logikájától idegen, hiszen a főprogram és a blank common változói kivételével alapértelmezésben egyik változó sem statikus, azaz a nekik lefoglalt memória az alprogramból kilépéskor felszabadul. Tudatos memóriagazdálkodásra is van mód az *allocate* és *save* utasítások révén. (A nem blank common-beli változók értéke úgy őrizhető meg, ha a blokk neve egy külön *save* utasításban szerepel.)

Az *equivalence* utasításnak új programokban nincs más létjogosultsága, mint a program olvashatóbbá tétele, pl. azzal, hogy egy include-dal beolvasott common-beli változók közül egynek-kettőnek „beszédes nevet" adjunk. E szerep viszont pont ugyanaz, mint a Fortran 66-ban volt. Példa:

```
common /a/ b(1000) ; equivalence (b(627),fontos_adat)
```

Egyébként a felhasználói adattípusok sokfélesége miatt az utasítás szabályai lényegesen szigorúbbá váltak, mint a korábbi Fortran-változatokban voltak (❑ g). Ezek lényege, hogy különböző típusú és fajtájú adatok összekeverése így általában tilos.

A kezdetiérték-adás lehetőségei nemcsak a típusdeklaráció új formáival bővültek, hanem a DATA utasítás keretében is. Skaláris változók, tömbök, tömbszeletek kaphatnak kezdeti értéket, akár ciklusutasítással is, például így:

```
integer, dimension (10000) :: nagy_vektor
real, dimension (10,0:2) :: matrix
data (nagy_vektor(i),i=1,10000,2) /5000*1/
data (nagy_vektor(i),i=2,10000,2) /5000*2/
data (matrix(i,1),i=1,10) /10*1.23/
```

A Fortran 77-től eltérően egy dupla pontosságú REAL változónak DATA utasítással adott kezdeti érték pontossága a számkonstans pontosságának felel meg, így a /1.23/ és /1.23d0/ két különböző kezdeti értéket jelent.

A Fortran 66 szövegkonstansai (Hollerith-konstansok) a Fortran 90-ben nem szerepelhetnek data utasításban. (Minden ilyen szerkezet egyszerűen átalakítható character típusú változókat tartalmazókra.)

Utasításfüggvények (❑ h)

Az utasításfüggvényekkel kapcsolatban csak egy előírás bővült a Fortran 66-hoz képest: a deklarációban nem kötelező a formális paraméter alkalmazása. A zárójelpár viszont kell, másképp a fordítóprogram skaláris változó értékadását tételezné fel. Tehát:

```
függvény-név ( [ formális-paraméter-lista ] ) =
kifejezés
```

ahol a kifejezés nem tartalmazhat a felhasználó által definiált operátort. A formálisparaméter-listán szereplő azonosítók lokálisak az utasításfüggvényre nézve. Globális változók is használhatók a kifejezésben. A függvény-név lokális az őt tartalmazó szegmensben. Ez azt is jelenti, hogy *external* utasításban nem szerepelhet, tehát más eljárásnak nem lehet paramétere.

A felhasználó által definiált típusok (❑ g)

A felhasználó által definiált típusok (derived type) a Fortran 90 belső (integer, real, complex, logical és character) típusai szerinti, valamint más felhasználói típusokból felépített struktúrák. Használatukhoz definiálni kell őket, majd a típusdeklarációk között fel lehet sorolni az érintett változókat. A definíció formája:

```
TYPE [ [ , hozzáférés ] :: ] típus-név
[ { PRIVATE | SEQUENCE } ]
típusdeklaráció
[ típusdeklaráció ]...
END TYPE [ típus-név ]
```

A hozzáférés csak modulokban definiált típusra adható meg: vagy PRIVATE lehet (a struktúra komponensei kívülről hozzáférhetetlenek), vagy PUBLIC (szabad hozzáférés). A SEQUENCE specifikáció azt írja elő, hogy a komponensek tárolási sorrendje egyezzék meg a felírási sorrenddel.

A típusdeklaráció tulajdonságként csak a DIMENSION és a POINTER kulcsszavakat tartalmazhatja, viszont lehet egy már definiált felhasználói típus deklarációja is. Kezdetiértékadás nincs értelmezve.

Példa a felhasználói adattípus definiálására, változók deklarálására és az értékadásra:

```
type csomopont
  integer ut_szam, kilometer_ko
end type
type falu
  character (len=20) falunev
  type (csomopont) falukozpont
end type
type (falu), dimension (1000) :: helysegnevtar
helysegnevtar(621) = falu('felcsut',811,38)
```

A struktúra egyes elemeire külön is lehet hivatkozni, az összetevők nevét %-jellel elválasztva. Például a 811-es út melletti településeket így lehet megkeresni:

```
if (helysegnevtar%csomopont%ut_szam==811) then ! stb.
```

Az egyszerűbb felhasználói típusok a tömbök általánosításának is tekinthetők, mégpedig úgy, hogy nem követeljük meg az összes tömbelem azonos (elemi) típusát. A fenti példát folytatva: a helysegnevtar első oszlopa a falunev charactervektorra, a második és harmadik oszlopa pedig az integer csomópont-mátrix első, illetve második oszlopára való hivatkozással lenne programozható Fortran 77-ben.

```
function falvak_tavolsaga(honnan,hova)
type (falu), intent (in) :: honnan,hova
type (falu) :: falvak_tavolsaga
  ! ket telepules tavolsagat adja meg az
uthalozati_terkepen
  ! megkeresett legrovidebb utvonalon
  ! az adat falvak_tavolsaga%csomopont%kilometer_ko-be
kerul
  ! falvak_tavolsaga%falunev = '????????????????????'
  ! falvak_tavolsaga%csomopont%ut_szam = -1
! lokalis valtozok deklaraciojanak helye
! eljarastorzs helye
end function falvak_tavolsaga
!
subroutine km_konverzio(numerikus,felhasznaloi_forma)
integer, intent (out) :: numerikus
type (falu), intent (in) :: felhasznaloi_forma)
numerikus=felhasznaloi_forma%csomopont%kilometer_ko
end subroutine km_konverzio
```

Az így definiált függvények a feldolgozóprogram további egyszerűsítését teszik lehetővé, a felhasználói típusra vonatkozó operátorok deklarálásával, ami viszont már a programszerkezeti utasítások közé tartozik.

Szondi Egon János

Hogyan jussunk A:-ról B:-re?

Többlépcsős bevezetés

Nem hagyta nyugodni szerkesztőségünket, hogy a különböző méretű floppymeghajtók A: és B: elnevezésének keveredése megnehezíti lemezmellékletünk automatikus indítását. Addig piszkáltuk a témát, míg most azután egyszerre két forró nyomra is rábukkantunk.

Állandó külső szerzőnk, Simay Endre István írt az Új Alaplap lemezmellékletéhez egy hasznos segédprogramot, pontosabban egy egész kis programcsokrot, tekintettel arra, hogy 1996-tól az A:-ról B:-re jutás fordítottja is aktuálissá válik, bizonyos esetekben pedig fontos lehet az eredeti állapot visszaállítása.

BALAP.BAT

A floppymeghajtók jelölését a lemezmelléklet valamennyi futtatható batch állományában szükség szerint A:-ról B:-re átíró A_BKONV.EXE program egyelőre nincs bekötve a szokásos ALAP.BAT indítóállományba, hanem az csak a BALAP.BAT nevű kiegészítésben szerepel.

Mindazok, akiknek 5,25"-es meghajtója B: jelzésű, és emiatt eddig átnevezési vagy átmásolási műveletekre kényszerültek, most a BALAP.BAT-ot elindítva „közvetlenül fogyasztható" lemezmellékletet kapnak.

Ha beválik, s ha a többség is erre szavaz, akkor „előléptetjük" egységes ALAP.BAT-tá, és használatával akár A:, akár B: jelzésű meghajtóból indítható lesz a lemezmelléklet. Ha ugyanis az A: meghajtó az 5,25-ös, tehát nem kell átírást végrehajtani, akkor a program intelligens módon nem változtat meg semmit. (A lemezindítás időigénye persze egy kicsivel több lesz.)

Azért, hogy a B: meghajtóra történt átírás után mégis eredeti formájában, tehát az A: meghajtót tartalmazó batch állományokkal lehessen archiválni a lemezmellékletet, vagy pedig egy másik gépen ismét az A: meghajtóból tudják azt futtatni, mellékeljük a B_AREGN.EXE programot is, amely visszaállítja a lemez A: meghajtóra hivatkozó előző állapotát.

SWAPAB.COM

Miközben a lemezátírás módozatait próbálgattuk, váratlanul jelentkezett egyik olvasónk, Delbel Dezső, hogy rábukkant egy apró programra. (Méghogy apró! Mikro: 148 bájtos! Annak is a fele szöveg! Ajánljuk a Microsoft és a hardvergyárosok figyelmébe.) Ezt a SWAPAB.COM-ot elindítva az A: meghajtóból egyszeriben B:, újra elindítva pedig ismét A: lesz. Mint egy billenőkapcsoló, úgy működik.

Ha ezt a programot sikerül úgy bekötni az ALAP.BAT indítóállomány elejére és végére, hogy az olvasót nem kell azzal zargatni, milyen betűjelű meghajtóba dugta a lemezt, akkor ez a megoldás is szimpatikus lehet.

Külön azonban máris minden további nélkül futtatható, ezért gyorsan rátettük a mostani lemezmellékletre. De „használat után" ne felejtsék el újra elindítani, hogy visszaállítsák a kiinduló helyzetet. És arra is gondoljanak, hogy a program egyetlen meghajtó esetén is „felcseréli" a betűjelzéseket!

Természetesen egyéb megoldásokat is szívesen fogadunk, várjuk olvasóink további ötleteit, illetve a most bemutatott két lehetőségre vonatkozó tesztelési tapasztalatait, észrevételeit.

Faklen Pál

Eredményhirdetés

az Új Alaplap közvéleménykutató kérdőívét beküldők nyereménysorsolásáról

Főnyeremény — IBM Aptiva minitorony, a Computer 2000-től:
Pajor Sándor, Oroszlány

II. díj — Psion Series 3a mobil számítógép, a Psion Kft-től:
Tóth László, Székesfehérvár

III. díj — OKI OJ 300c színes nyomtató, az OKI Kft-től:
Ávéd Zoltán, Ászár

IV. díj — A magyar nyelvű Windows 95 első példánya, a Microsofttól:
Bende Imre, Szombathely

1-1 db Microsoft Works 3.0, a Microsofttól:
Balogh Lászlóné, Bátonyterenye; Bolgár Gábor, Budapest XVIII.

1-1 db Microsoft Pro Mouse, a Microsofttól:
Pulai Zsolt, Nyul; Ilyés László, Budapest XVI.

1-1 db Microsoft Home Mouse, a Microsofttól:
Mihály Miklós, Szolnok; Csordás László, Budapest XVII.

1-1 db Microsoft Scenes, a Microsofttól:
Pém Mihály, Miskolc; Dankó János, Debrecen

Microsoft Cinemania 95, a Microsofttól:
Ruttkay János, Budapest

J.F.K. Assasination, a Keszo Kft-től:
Hidvégi András, Budapest XI.

Kiplinger's CA-Simply Money, a Keszo Kft-től:
Fehér Lászlóné, Budapest X.

The Lawnmover Man, a Keszo Kft-től:
Frühwald Ferenc, Dunaújváros

Sing-Along With, a Keszo Kft-től:
Holczinger József, Győr

1-1 db Kodak írható CD-ROM, a Kodak Kft-től:
Csonki Zoltán, Békéscsaba; Makray Róbert, Budapest XXI.; Megyik Attila, Szarvas; Tóth György, Gyöngyös; Arany Tamás, Szombathely

1-1 db 2 éves előfizetés az Új Alaplapra:
Szilvási Gábor, Budapest X.; Takács Ferenc, Kunszentmárton; Kereki István, Szikszó

3-3 doboz 360-as KAO lemez:
Bodnár Erika, Budapest; Bartha Balázs, Dunaújváros; Süveges József, Sajószentpéter; Czibik Tamás, Budapest X.; Nagy Sándor, Gyula; Kovács Miklós, Érd; Kiss Csaba, Veszprém; Borbély László, Kecskemét; Süveges Judit, Sajószentpéter; Faragó Tibor, Budapest IV.

1-1 doboz 1,2 MB-os MIC lemez, a Teta Kft-től:
Szatmári Éva, Budapest; Fehér Olga, Budapest X.; Kámán Attila, Debrecen; Édelmayer Mihály, Tápiószele; Huszár Róbert, Budapest VIII.; Pruch Mihály, Budapest XV.; Vajda Károly, Budapest III.; Németh Károly, Ajka; Gácsi József, Budapest XXIII.; Varga Zsolt, Pécs; Csaszny Márton, Budapest XX.; Vinnai Gyula, Miskolc; Solti Imre, Budapest XXII.; Kovács Barnabás, Budapest XXII.; Fehér László, Budapest X.

CorelDraw egéralátét:
Szincsák István, Salgótarján

Ifabo '96 egéralátét:
Megyeri Zsolt, Veszprém

IBM OS/2 egéralátét:
Boros Attila, Püspökladány

Lotus SmartSuite egéralátét:
Nagy Tibor, Budapest II.

Genius egéralátét:
Kovács László, Budapest III.

Hewlett-Packard kéztámasztó:
Hajdú Tibor, Budapest I.

A NEM KERESKEDELMI CÉLÚ EGYÉNI HIRDETÉSEK KÖZLÉSE INGYENES

A kereskedelmi célú apróhirdetések tarifája gépelt soronként (azaz 60 karakterenként) 300 forint. A terjedelem alapján így kiszámított összeget kérjük az Új Alaplap Kiadói Kft számlájára átutalni (OTP, 218-98017 / 501-017164-7), vagy postautalványon a kiadó címére elküldeni (1538 Budapest, Pf. 571), és feltüntetni, hogy „Új Alaplap, apróhirdetés". A befizetést igazoló szelvény másolatát — a hirdetési szöveggel együtt — a szerkesztőséghez (a kiadóéval azonos címre) küldjék el.

A szerzői jogokat sértő szoftverhirdetéseket nem tesszük közzé. (Lásd erről bővebben 1994. januári számunkat.)

Bármilyen típusú szöveg fordítását vállalom angol-magyar, magyar-angol nyelvre, illetve kiadvány látványtervezését, szerkesztését is. Cím: Lachner Zoltán 1195 Budapest XIX., Jáhn Ferenc u. 14/a. Telefon: 157-0308.

Objektumorientált programozás **Clipperben:** Objects 2.0. Kérésre tájékoztatót küldök. Cím: Szűcs János, 4400 Nyíregyháza, Vasvári Pál u. 37. Tel.: (42) 313-568 vagy 312-222/1382-es mellék.

Adatmentés CD-re, streamerre; winchesterről, floppyról. Ugyanitt beszerzési tanácsadást, hálózattervezést és programkészítést is vállalok. Cím: Kovács Lajos, 1031 Budapest III., Vízimolnár u. 10. IV/33.

Stúdiómban megbízhatóan, ellenőrzötten lefordítom angol, német, francia és magyar nyelvről/nyelvre műszaki és közgazdasági folyóiratok cikkeit, hardver- és szoftverleírásait. Áfás számlát állítok ki. Cím: Szász György, 1035 Budapest III., Kórház u. 25. Tel.: 168-4874.

Alaplapcsere, memória-, winchester- és floppybővítés a helyszínen. MegaSoft. Telefon: 295-5085.

Eladók régi és új számítástechnikai újságok. Érdeklődőknek listát küldök. A lapok ára 150 Ft/db + postaköltség! Cím: Kocsis Zoltán, 5900 Orosháza, Dózsa Gy. út 13.

Főiskolai diák **korrepetálást** vállal számítástechnikából kezdőknek, pótvizsgára szorultaknak. Ugyanitt programcsere is lehetséges. Cím: Kovács Gábor, 3502 Miskolc II., Pf. 83.

Keresek keskeny nyomtatót (A4-es levelek írásához). Cím: Kellner Miklósné, 8700 Marcali, Széchenyi u. 18. Tel.: (85)312-213.

Eladók PC-X, Guru, PC-Guru, CoV, Cool magazinok, valamint PC World-ök és Új Alaplapok lemezmelléklettel együtt. Cím: Pető László, 2541 Lábatlan, Piszkei ltp. 7/4.

VT64 mikroszámítógéphez keresem a VT-DOS rendszerlemezét, mert az eredeti tönkrement. Egyéb programok is érdekelnek. Aki tud, kérem, segítsen, mert a gép így használhatatlan. Cím: Bagó István, 9081 Győrújbarát, Kert u. 17. Tel.: (96)355-312.

Eladó 2 db 5.25" HD floppy drive és 1 db 3.5" HD floppy drive 4000 Ft/db áron, vagy színes SVGA monitorra cserélhető ráfizetéssel. Érdeklődni lehet a 186-4026 telefonszámon az esti órákban.

Compfairen megjelenő hazai fejlesztésű szoftvercsomag kézikönyvében — amely hiánypótló nyomdai, DTP, hardverdiagnosztika programokat tartalmaz — két oldal **hirdetési felület eladó.** A programok és a kézikönyv tördelt változata megtekinthető. LEZLISOFT Computer Graphics. Tel.: 163-1771.

Számítástechnikából oktatást és korrepetálást, valamint beszerzési tanácsadást és programkészítést is vállalok. Tel.: 169-4146.

Eladó 386-os számítógép, színes VGA monitorral, 1,44 és 1,2 MB floppymeghajtóval, winchesterrel. Ugyanitt hangkártya is. Telefon: 295-5526.

Ingyen több megabájt jogtiszta (shareware, freeware) PC program! Küldj egy felbélyegzett válaszborítékot a listáért. Cím: Tilly György, 1139 Budapest, Rozsnyai u. 5.

PC-hez **Disney hangmodulok** olcsón eladók. Válaszborítékért tájékoztatót küldök. Cím: Varga Zoltán, 9731 Kőszeg, Pf. 9.

Keresek ACHIVE 5945C típ. streamerhez vezérlőkártyát, szoftvert. Cserébe tudok adni VL IDE kártyát (4 wincsit, 2 floppyt vezérel). Garancia van rajta, új, használva nem volt, vagy 3 db CD-ROM-ot. Cím: Kiss János, 5000 Szolnok, Ispán kr. u. 3. X/6. Tel.: (56)379-490.

E számunk hirdetői

Kisfilmek a nagyvilágból

IDE nekem az oroszlánt!

Mostani számunk Könyvespolc rovatát egyszerre „több ablakban" lehet olvasni. A könyvek ugyanazzal a témával foglalkoznak, a Windows-felületen futó alkalmazások programozásával, de mindegyik más megközelítésben. Maguk a recenziók is olyannyira más nézőpontból születtek, hogy az egyik kifejezetten elkívánkozott innen a hónap témájának összeállításába, ahol a Windows-világ általános, néha egészen filozofikus felvetései kaptak helyet. A Windows „lelkivilágáról" tehát a 11. oldalon olvashatunk, „testi valóságáról" pedig inkább itt...

Benkő Tiborné—Benkő László—Tóth Bertalan—Kiss Zoltán:

Windows felhasználói programok Borland C++ környezetben

ComputerBooks, 1993 (2. kiadás)
240 oldal, 691 Ft (lemezmelléklettel)

Aki először fog Windows programok készítésébe, nem bánja meg, ha a szerzők által bemutatott (és az IDE-vel kiválóan megtámogatott) utat járja (IDE = Integrated Development Environment, a Borland cég integrált fejlesztőrendszere). Bármilyen csábító lehetőségeket kínál is az objektumorientált programozás a Windows programok készítéséhez, kétszer is meggondolandó, hogy érdemes-e az objektumorientált technika „metaszervezési" problémáival elvonni a figyelmet a Windows programok működésének más szinten jelentkező szervezési problémáiról. Később, a Windows működésének alaposabb megismerése után talán neki lehet vágni az OOP-vel támogatott fejlesztésnek is. Az első lépéseket azonban mégiscsak könnyebb megtenni, ha egyszerű DOS programfutások és szerkesztések egymásutánjává szelidül a programok elkészítése. A lényegen az nem változtat, hogy a Windows programok moduldefiníciós állományát is hozzá kell szerkeszteni a modulokhoz, majd az erőforrásokat is bele kell építeni az .EXE programba.

A BC rendszerek (amelyek a Turbo C rendszer egyenes ági leszármazottai) két igen fontos tulajdonsággal bírnak. Egyrészt Windows alá is tudnak programokat fejleszteni, másrészt fel tudják használni az OOP gazdag lehetőségeit. E könyv példái mégis csak az első lehetőséget használják ki, és szinte kizárólag közönséges C nyelven íródtak, egészen a program futtatásáig nem is igénylik a Windows támogatását. Nem véletlenül. Valóban, biztonságosabb lehet, ha magát a fejlesztést hagyományos körülmények között végezzük, és csak a futtatást kérjük a Windowstól. (Megjegyzendő, hogy a BC IDE azért felkínálja a Windows alatti DEBUG futtatásával a segítséget a program belövéséhez.) Ezt az egyszerűbbnek tűnő utat tudatosan vállalták fel a könyv szerzői, és igen meggyőzően mutatják be, hogy nem is volt rossz választás. Jól kidolgozott példák sorozatán keresztül magyarázzák az egyre bonyolultabb Windows alkalmazói programok készítésének mesterfogásait. A példák filmkockák módjára követik egymást, jó lehetőséget nyújtva az apró eltérések alapos megfigyelésére.

Sok hasznos ismeretet közölnek közben a Windows programnak az előállításáról és működéséről is, és elemzik a különböző memóriamodellek sajátosságait. Részletesen bemutatják a Whitewater Resource Toolkit szerkesztőrendszert, amelynek segítségével könnyűszerrel lehet szerkeszteni, módosítani a Windows programok számára megadott erőforrásokat. Megtanulhatjuk a könyvből, miként lehet a C és C++ nyelven írt modulokat Assembly nyelven készült modulokkal „keverni", sőt a könyv függelékében a BC++ Classlib osztályának felépítését, és az egyes osztályokhoz tartozó függvényeket is megismerhetjük a felhasználáshoz szükséges mélységig.

Használati utasításként érdemes megjegyezni, hogy ez a könyv is az olyan művek közé tartozik, amelyeket az illusztrációk végignézésével érdemes kezdeni. Szükségeltetik hozzá a Borland C++ integrált fejlesztőrendszerének legalább a 2.0 verziószámú változata, és persze egy installált 3.x Windows. Aki ezekkel rendelkezik, rögtön felviheti a lemezmelléklet önkibontós példaprogramjait a merevlemez újonnan nyitott könyvtárának alkönyvtáraiba. Egyetlen könyvtárba nem érdemes összekeverni a példákat, mert úgy nehezebb áttekinteni, hogy mikor mit csinálunk. (Egy kis batch-program egyébként valamennyi programot kibontja nekünk.) Válogatni is lehet a kipróbálandó programok között: tessék megnézni a tartalom.txt szövegfájlt.

A példák kipróbálásához a megfelelő alkönyvtárból indítsuk el a Borland integrált fejlesztőrendszerét a bc paranccsal. A kipróbálásban lényegében rábízhatjuk magunkat a könyv szövegére, de azért nem árt egy kis óvatosság az opciók beállításánál. Például megnézhetjük, hogy milyen könyvtárakból veszi az IDE az Include fájlokat, hogy a függvényhívási konvenció hogyan van beállítva, stb. Zavart okozhat, hogy a .PRJ kiterjesztésű fájlok előre el vannak készítve a példák segédállományai között, de egyáltalán nem biztos, hogy a számunkra megfelelő tartalommal. Jobb, ha ezeket egyszerűen kitöröljük, s magunk készítünk helyettük újakat a könyv útmutatásai szerint. A megadott .PRJ fájlok ugyanis nem érhetők el például akkor, ha más elnevezésű könyvtárban van a Borland rendszerünk, mint amit a .PRJ fájlokba „beledrótoztak", ekkor ugyanis a lefordítandó és az inkludálandó fájlok nem találják meg egymást.

Óhatatlanul felmerül mindenkiben a kérdés, hogy miként lett Borland C a Turbo C-ből. Nos, a TC 2.0-s verziója után a Borland cég gyorsan egymás után bocsátotta ki az IDE objektumorientált irányba továbbfejlesztett változatait. A TC++ még akkoriban jelent meg, amikor a 3.0-s Windows. Az újabb fejlesztések, amelyek már a Windows-környezetre is felkészültek, azzal hívták fel magukra a figyelmet, hogy új, bár régről ismert néven Borland C-ként kerültek a piacra. A BC rendszerek másik közös jellemzőjeként érdemes megemlíteni, hogy sokkal jobban ki vannak stafírozva, mint a TC, a „szegény rokon".

A BC 3.1 ennek is több mint a kétszeresére nőtt: 15 lemezre. Itt már a könyvtárak forrásnyelvű változatait is megtaláljuk, új könyvtárakat. Köztük igen sok Object Win-



dows könyvtárat (OWL) és futásidejű könyvtárat (RTL), és töméntelen sok demóprogramot. Külön csemege a WinSpector rendszer, amely segédprogramjaival együtt a Windows alatt meghalt programok Unrecoverable Application Errorjainak post mortem vizsgálatához nyújt segítséget. (Ismeretes, hogy a Windows alatti programfejlesztés sokkal nagyobb elővigyázatosságot kíván: itt egy programhiba gyakran az egész rendszer elszállását okozhatja.)

Jelenleg a BC 4.5-ös verziója a legfrissebb. Nem okozott meglepetést, hogy továbbra sem csökkent a terjedelem...

Visszatérve a könyvre, igazán kár, hogy a szerzők új könyvük megírásán dolgozva (amely már OWL-környezetben mutatja be a Windows alkalmazói programok fejlesztését) nem sok gondot fordítottak e „régi" könyvük új kiadásának előkészítésére. Így fordulhatott elő, hogy helyenként félbe hagyott, sőt értelmetlenre sikeredett mondatok is maradtak benne, főleg a 4. fejezet tájékán. A könyv helyesírását főleg a rengeteg indokolatlan különírás rontja le (nyomó gomb, forrás program, ablak méret, üzenet sor stb.). Két helyen még a suk-süközés jelei is megmutatkoznak... Egészében véve a könyvet az alapos, világos magyarázat, a logikus okfejtés jellemzi. Négy szerző és egy lektor viszont már a hibák kigyomlálásáról is gondoskodhatna.

Benkő Tiborné—Moré Gábor:

Object Windows

Windows-alkalmazások gyorsan, könnyedén! Objektumorientált Windows programozás Borland C++ rendszerben
ComputerBooks, 1993
396 oldal, 979 Ft (lemezmelléklettel)

Ez a könyv éppen a másik oldalról közelíti meg a Windows felhasználói programok fejlesztését, mint az előző: nem kitágítani igyekszik azt a hagyományos gondolkodásmódot, amely a DOS alatti programfejlesztést jellemezte, hanem inkább meg szeretne szabadulni tőle. A támogatást ehhez is a Borland szoftvereiben találják meg a szerzők. A „Borland C++" néven egyre nagyobb népszerűségnek örvendő fejlesztőrendszerről különösen a 3.0-s verziótól kezdve mondható el, hogy egészen kiváló eszközökkel segíti a Windows felhasználói programok fejlesztését. Igaz, hardverkövetelményei is jelentősek, de még mindig szerényebbek, mint a Microsoft SDK fejlesztőrendszerének (SDK = Software Development Kit).

A BC++ számos tekintetben igazodik ugyan a Microsoft-fejlesztésekhez (különösen ami a csatlakozási felületeket illeti), de kiegészíti ezeket saját felhasználóbarát eszközeivel. Nem ritkaság, hogy a Microsoft nem kellően kiérlelt, nem eléggé kidolgozott termékeket dob piacra, s keveset törődik vele, hogy mibe kerül ezek gargantuai étvágyának csillapítása. A Borland fejlesztőgárdája összehasonlíthatatlanul kisebb, de ez a jól összeszokott csapat mintha többet adna a mundér becsületére. Róluk elmondható, hogy termékeik gyakran szerényebb feltételek mellett is többet nyújtanak, s ami nem utolsó szempont, kellemesebb környezetet biztosítanak a felhasználóknak.

A könyv a BC++ gazdag lehetőségei közül azokat igyekszik bemutatni, amelyek kész lehetőségként állnak a felhasználók rendelkezésére az Object Windows Library (OWL) könyvtárakban. Mindezt belehelyezi egy kimondottan felhasználóbarát környezetbe. Ez a Borland C++ Application FrameWorknek nevezett rendszer tulajdonképpen a Turbo családból jól ismert IDE windowsos megfelelője. A rendszer fő előnyének azt tekinthetjük, hogy egyetlen keretben fogja össze a programfejlesztés és -futtatás valamennyi lépését, s az opciók beállítása alapján automatikusan elvégzi, amit csak lehet, a különböző ellenőrzésektől kezdve a bonyolult futássorozatok megszervezéséig.

Az OWL könyvtárak használata mentesíti a felhasználókat a rutinszerűen megfogalmazható részletfeladatok hosszadalmas kódolási is belövési terheitől, s lehetővé teszi nekik, hogy ahol az megoldható, ott előregyártott elemekből építkezzenek. A Windows-alkalmazások területén ennek különösen nagy jelentősége van. A biztonságosan felhasználható, hibátlan alkatrészekből a felhasználó könnyebben és gyorsabban készítheti el saját tervezésű bonyolultabb rendszereit is. Az alkatrészek hibái a Windows esetében súlyosabb következményekkel járhatnak, mint a DOS egyszerűbb futtatási körülményei között. A Windows-futások annyiféle komponens összehangolt együttműködését kívánják meg, hogy nagyon könnyű hibázni. Apró programhibák is az egész rendszer összeomlásához vezethetnek, s ilyenkor nyomtalanul eltűnik az a kritikus állapot is, amely magát a robbanást kiváltotta.

Az talán túlzás, amit a könyv címe sugall, hogy gyorsan, sőt könnyedén fogja tudni Windows-alkalmazásait elkészíteni az, aki ezt a 400 oldalt áttanulmányozza. Annyi viszont túlzás nélkül elmondható, hogy a legtöbb esetben tudni fogja, mihez milyen szerszámokra van szüksége, és miért hová kell nyúlnia. Kiválóan szerkesztett, jól illusztrált árjegyzéket nyújt ez a könyv, amelyben típusmegoldások találhatók a típusproblémák rendkívül széles skálájára. Az előző könyvtől eltérően, amely filmkockaszerűen egymásra épülő feladatok sorozatán keresztül vezetett be — talán lassabban, fokozatosabban — a Windows programozás rejtelmeibe, itt egymástól eltérő feladattípusokra kaphatjuk meg az OOP szellemében fogant frappáns megoldást. Ki-ki megkeresheti a feladatok között azt a témakört, amely közelebb áll érdeklődéséhez, s a bemutatott módszer variálásával viszonylag kis erőfeszítéssel már maga készíthet számos hasonló probléma megoldására Windows programot.



A Windows programozásának van egy, ma már hagyományosnak tekinthető módszere: az Application Programming Interface (API) függvényhívások alkalmazása, amelyek definícióit a WINTYPES és WINPROCS modulok tartalmazzák. Aki részleteiben is meg akar ismerkedni a Windows programozás nyalánkságaival, az aligha kerülheti el, hogy egyszer át ne rágja magát ezek működésén. De ha közelebbről kipróbálta már, hogy mennyi mechanikus beírással jár akár csak a függvények korrekt paraméterezése, aligha vágyakozik rá, hogy ezeket minden esetben saját magának kelljen elvégeznie. Valóban, ez az a tevékenység, amely roppant mértékben leegyszerűsödik, ha az OWL könyvtárra és a Borland keretrendszerére hagyatkozunk. A mechanikusan megadható paramétereket a rendszer maga kitölti, a hiányzókra pedig dialógusablakokban rákérdez. Az OWL tehát felöleli a Windows teljes API apparátusát, de jótékonyan eltakarja a részleteket.

Ez a könyv tehát a gyakorlat oldaláról mutatja be a Windowsban való programozást. Ezt viszont imponáló alapossággal és magabiztossággal végzi, és az elméletből is ad annyit, amennyi a gyakorlat megértéséhez elengedhetetlen. Jól összefogott és magvas összefoglaló olvasható a Windowsban való programozásról és a C++ nyelvről ennek a könyvnek a bevezető fejezeteiben és függelékében. A könyv többi része maga a gyakorlat: az ObjectWindows-alkalmazások bemutatása a példák tükrében. Sokat segítenek a megértésben a könyv illusztrációi is. És természetesen, akinek módja van rá, az „élőben" is kipróbálhatja a könyv gazdag illusztrációs példaanyagát. Egyébként a példák közül elég soknak a Windows alatt futtatható EXE programja is megtalálható — ezeken keresztül az is érzékelheti az eredményt, aki egyelőre még nem tudta beszerezni magát az „alaprendszert".

Külön érdemes kiemelni, hogy aki a fejlesztésnek ezt az útját választja, olyan eszközfüggetlen saját grafikát építhet be rendszerébe, amelynek megjelenítése semmilyen szabványos monitoron nem okoz problémát, s emellett igen gazdag kész grafikus könyvtárból is válogathat.

V. Nagy Edit

Ismét Alaplap Posta!

Az egykor volt Alaplap Posta hagyományait felélesztve — kísérleti jelleggel — vállalkozunk arra, hogy a kiválasztott CD-ket postai utánvéttel eljuttatjuk Önhöz. (Az utánvét díja a megrendelőt terheli.) Megrendeléséhez használhatja a középső lemezvédő kartonból kivágható válaszlevelezőlapot, vagy annak fénymásolatát! (Mivel a forgalmazó Automex az árváltoztatás jogát fenntartja, az itt közölt árak csak tájékoztató jellegűek!)

1. The Magic School Bus

Gyermekeknek készült ez az emberi testet bemutató, játékos ismeretterjesztő program, amelyben szerveink működése, elhelyezkedése vehető szemügyre. Látványos grafikus magyarázatokat, szemléletes ábrákat vonultat fel, játszva tanítja meg az emberi test felépítését. (14 590,–)

2. Apogee's Greatest Games

A legjobbak közül került fel 15 játék erre a CD-ROM-ra. Néhány cím a választékból: Raptor, Commander Keen, Secret Agent, Duke Nukem I-II., Word Rescue. (6790,–)

3. Fruit Machine

Rengeteg humoros megoldással és grafikával fűszerezett játék, mely a nyerőgépek világába kalauzol el. (3890,–)

4. Stack Up

Íme a legújabb képviselője a táblás logikai játékoknak. Látványos grafikai megoldások, szórakoztató és gondolkodtató kikapcsolódás bármely korosztálynak. (3890,–)

5. World War II.

A II. világháború 2100 eseményét dolgozták fel CD-ROM-on, több mint 900 fényképpel. 140 harci térképpel, eredeti film- és hangfelvételekkel illusztrálva. Végignézheti egy kiválasztott esemény történéseit, magyarázatokkal kiegészítve. Eredeti felvételek a Pearl Harbour-i bombázásról, a holocaustról és még sok másról. Híres tábornokok taktikáját ismerhetjük meg lépésről lépésre. (9790,–)

6. Arthur's Birthday

Interaktív mese gyerekeknek. Játszva tanulhatják az angol nyelvet. Rengeteg ének, mese hangzik el a lemezről. (9790,–)

7. PC Technician

Biztosan tapasztalta már, hogy gépe nem működött teljesen hibátlanul. Vagy elvitte szervizbe, vagy nem tett semmit, csak bosszankodott. Ezen a lemezen olyan programokat talál, amelyekkel diagnosztizálhatja az esetleges problémákat. Lekérdezhet olyan információkat gépéről, amelyek fényt deríthetnek a hiba okaira. Sok hasznos felhasználói program és egy interaktív ismertető van a lemezen, amely bemutatja napjaink legújabb technikáit a multimédiában. (3890,–)

8. Countries of the World

A Countries of the World hatalmas adatbázis a világ országairól, népeiről. Az alábbi területekről kérhet információkat: városok, földrajzi adatok, népek, nyelvek, politikai viszonyok, gazdaság, követségek, népszokások, ünnepek. (9790,–)

9. Leonardo

Ez a multimédia CD-ROM visszavezet a messzi múltba, ahol a századok legnagyobb látnok-géniusza. Leonardo da Vinci találmányait fedezhetjük fel a maguk történelmi környezetében. A program Macintoshon és Windows alatt egyaránt fut. (8090,–)

10. Cinemania

Összefoglaló a világ filmgyártásáról digitalizált video- és audiófelvételek alapján. Kitűnő szórakozás mindenkinek, aki szívesen tölti idejét kedvenc filmjeinek felidézésével. (8090,–)

11. Alpha Five for Windows

Egyszerűen és gyorsan szeretne adatbázist építeni? Ebben lehet segítségére ez a CD-ROM. Kevés gyakorlattal is eredményes lehet bárki, ha él a program által nyújtott lehetőségekkel. (Windows 3.1, 8 MB RAM, VGA monitor, CD-ROM drive.) (19 490,–)

12. Art Gallery

Ez a CD-ROM bemutatja és rendszerezi a Londoni Nemzeti Galéria művészeti alkotásait. Festmények, adataik, történetük, a festők életrajzai, összehasonlító elemzések; mindezek egy programba gyűjtve. (12 990,–)

13. Street Atlas USA

Komplett utcatérkép az USA összes városáról, vidéki területeiről. A legnagyobb részletességgel mutatja be az úthálózatokat, a legkisebb mellékutcákat is beleértve. Pontosság, egyszerű használhatóság jellemzi. (22 690,–)

14. Desktop Publisher's Dream

Ez a gyűjteményes CD-ROM több száz clip-artot és betűkészletet (fontot) tartalmaz. Az anyag könnyű kezelését egy jól áttekinthető füzet biztosítja, amely nyomtatott formában jeleníti meg a lemez anyagát. (6790,–)

15. Icons

Aki valaha is dolgozott Windows rendszerben, tudja, mi az, hogy ikon. Ez az apró kis ábra szimbolizálja a program jellegét. Ezen a lemezen 4000 ikon található, ami beilleszthető saját programjainkba vagy már meglévő ikonjaink helyére. (6790,–)

16. Windows Shareware

Ezen a lemezen 4000 válogatott shareware program található a Windows grafikus felhasználói felületre. (2090,–)

17. Barangolás az állatvilágban

A Föld állatvilágának néhány jelentős képviselőjét 325 színes képpel, másfél óra hanganyaggal és sok oldalnyi leírással mutatja be. A program Windows alatt fut a legjobban 800x600-as felbontásban, 256 vagy true color (16,8 millió) színben. Az enciklopédiában található állatok képei kinagyíthatók teljes képernyő nagyságúvá. 250 állatról tartalmaz fontos információkat a lemez. Az állatok csoportosíthatók a felhasználó által kiválasztott szempontok szerint. A lemez érdekessége, hogy a program help-rendszere egyedi megoldással készült. A megszokott módtól eltérően nem egy hosszú leírás jelenik meg ha valamit nem tudunk, hanem narrátor segít az információdásban. Ezzel a módszerrel az olvasni nem tudó gyerekek is tudják használni a programot. A lemez célja az általános ismeretek gyarapítása, nem pedig az egyetemi vizsgákra való felkészítés. E „hangos képeskönyv" minden korosztálynak ajánlott. (4990,–)

18. Budapest CD

A világ első 5 nyelvű információs, szórakoztató CD lemeze Budapestről turistáknak, nyelvtanulóknak, iskolásoknak. Budapest szerelmeseinek, egyszóval mindenkinek. A CD tartalmazza a főváros nevezetességein kívül idegenforgalmi, éttermi, kulturális, szórakozási, sportolási lehetőségeit is sok száz fényképpel, videóval, szöveggel és hanganyaggal. A CD-n megtalálható 1:30000 léptékben Budapest térképe az összes látnivaló feltüntetésével, keresési lehetőséggel. (5790,–)

19. Picdic

Magyar készítésű nyelvoktató CD-ROM lemez. Kiknek ajánlható? Kezdőknek, haladóknak, felnőtteknek és gyerekeknek egyaránt, vagyis mindazoknak, akik a szótanulás fáradságos munkáját szeretnék hatékonyabbá és könnyebbé tenni a számítógép felhasználásával. 82 témakör a madaraktól az űrutazásig, 200 színes kép, 5000 szó és kifejezés. Emberi beszéd, keresés, tallózás, tesztek... A lemezhez egy tankönyv is jár, mely a lemez anyagát tartalmazza. Angol, német és francia változata is megjelent. (6390,–)

Válasszon a mi válogatásunkból!

**Új Alaplap Kiadó,
1538 Budapest, Pf. 571**

Video-boszorkánykonyhában

Az illusztrálhatóságot tekintve vannak a számítástechnikában elég hálátlan szakterületek, ezzel szemben olyanok is akadnak, amelyeknél a szavakkal történő leírás tűnik szinte megoldhatatlannak, annyira szükség lenne a közvetlen vizuális élményre. Ezt a tapasztalatot igen intenzíven élhették át azok, akik a Creative Engineering meghívása alapján elmentek a Flame video-editáló rendszer bemutatójára, a Silicon Graphics magyarországi bemutatótermébe.

Persze nem gondolhatjuk, hogy amit láttunk, az hamarosan „közfogyasztási cikk" lesz. Egyelőre jó lenne, ha a tévések, a reklámfilmesek és más hasonló profik egy része rendelkezne ilyen eszközök használatának lehetőségével, de az angol „filmszerkesztő" bemutatója nagyon elgondolkodtató lehet a többi szakmabelinek is.

Azt már régen megszoktuk, hogy a zeneszámokat hangszerenként külön rögzítik, utána egymásra másolják az énekkel együtt. A hanganyaggal sok más manipulációt is lehet végezni, amíg csak a végleges hangzás ki nem alakul. Azt is tudtuk, hogy a videotechnikának mindenféle fantasztikus lehetőségei vannak a különböző effektusok, átmenetek, torzítások létrehozásában. Most viszont azt láttuk a Discreet Logic bemutatóján, hogy a Flame nevű programmal a Silicon Graphics gépén úgy lehet kezelni a videofelvételeket, ahogy gyerekkorunkban a gyurmával játszottunk. Sőt, úgyabbul!

Ezzel a digitális filmszerkesztéssel bármilyen effektust sokkal finomabban, sokkal rövidebb idő alatt lehet létrehozni, mint a hagyományos videotechnikával. Az pedig, hogy egy videofelvételen akár pixelenkénti precizitással végezhető el bármilyen retusálási feladat, majd a kész mű újra rögzíthető videoszalagon vagy filmen, a beavatkozásnak pedig tulajdonképpen semmi nyoma nem marad, igen érdekes kérdéseket vet fel a film dokumentumértékéről, a dokumentálás újrafogalmazandó kritériumairól.

Amikor az égi műholdas csatornákat nézzük, már ma sem mindig lehetünk biztosak abban, hogy fizikailag bebútorozott stúdiót látunk-e magunk előtt, vagy a bemondók székén és asztalán kívül minden tárgy csak virtuálisan van jelen. (Még a pohár víz is?!) Később könnyen megtörténhet, hogy helyszínek és szereplők képelemekből digitálisan megkomponálva kerülnek össze, pedig valójában fizikailag távol vannak egymástól. A (manipulációs) lehetőségek beláthatatlanok.

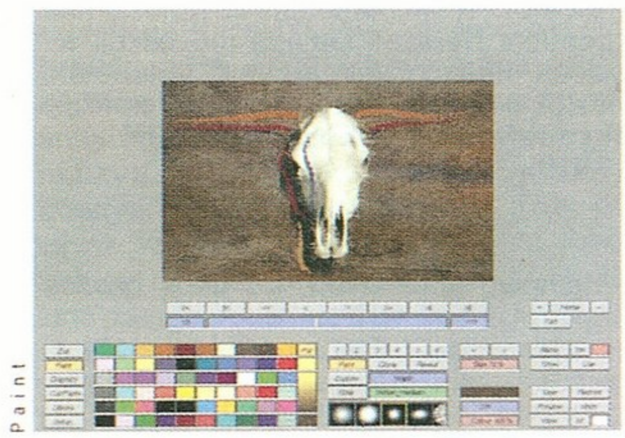
Szóval ilyen gondolatokat ébresztett bennünk ez a bemutató, és másoknak is csak azt ajánlhatjuk, hogy ha egy kiállításon vagy kisebb szakmai bemutatón alkalmuk nyílik megismerni e technikai boszorkánykonyhával, ne szalasszák el a lehetőséget. Az egérközelbe kerülés pedig azoknak is élmény lesz, aki előtte hagyományos technikával soha nem editáltak videoképsorokat.

Magyar dokumentumot magyarul

A Nádor Rendszerház forgalmazza az Optika Imaging Systems Incorporated által készített elektronikus dokumentumkezelő rendszereket. Ezek a feladat természetéből fakadóan az e rendszereknél már megszokott felépítésűek: nagysebességű szkennerek végzik az információ beolvasását, nagykapacitású tárolóegységek szolgálnak tárolásra, az adatvisszakeresés pedig már csak (jó) szoftver dolga.

Az Optika-disztribúcióval (DokuMagic, FilePower, ImageFiler, ImageFinder) gyakorlatilag teljessé vált a Nádor rendszerház fegyvertára: az erős Compaq-, Novell-, Micro-

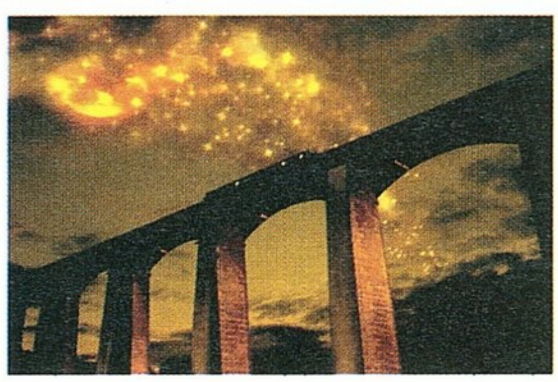




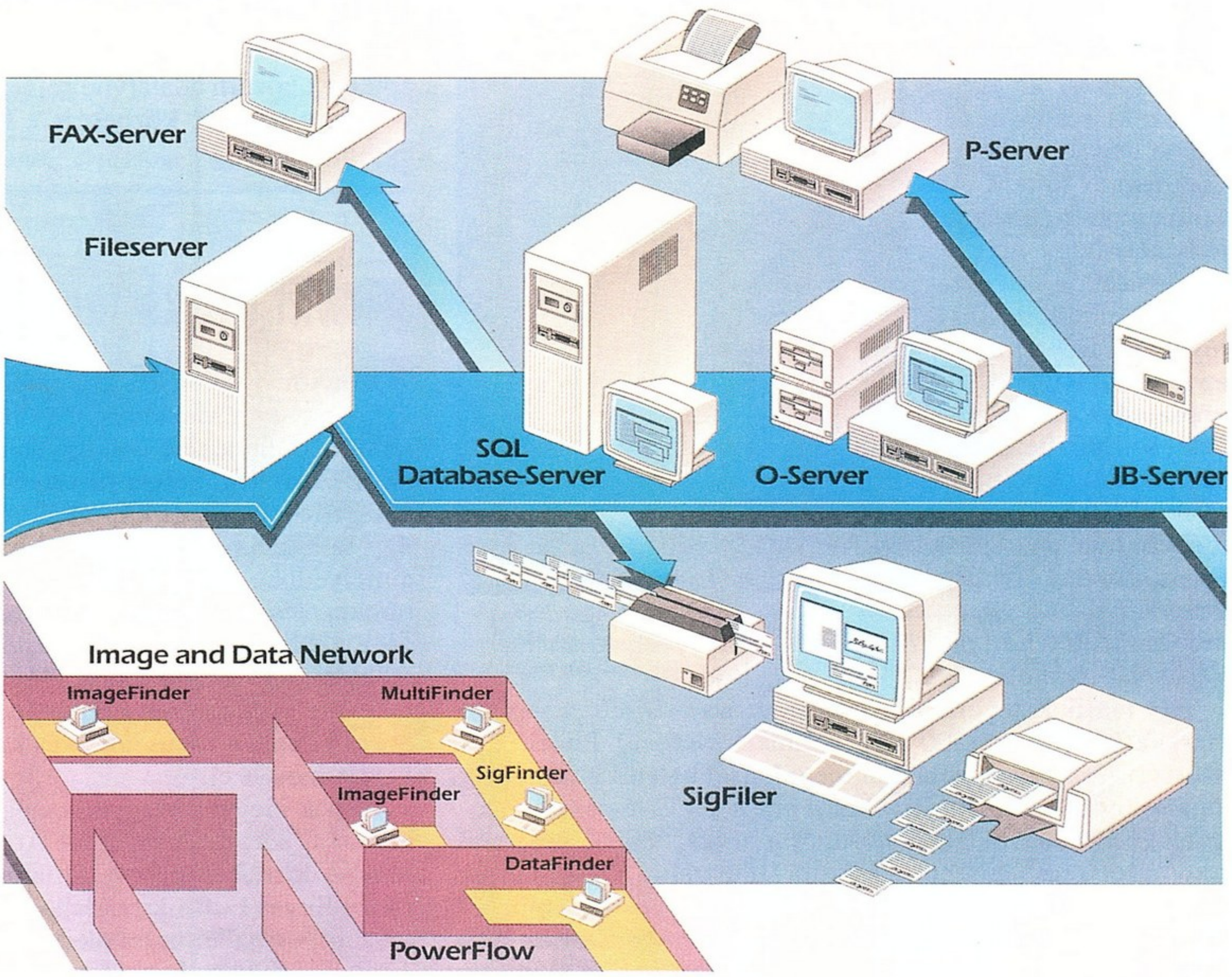






soft-, HP- és PSI-kapcsolat eddig is lehetővé tette a komplex megoldások kínálatának kialakítását, és most már egyes különleges nagyfelhasználói igények elé sem gördülhet akadály.



Az elektronikus dokumentumkezelő rendszerek olyan felhasználóknak ajánlhatók, akik — egészében vagy részben — szeretnének megszabadulni a munkájukat kísérő papírhegyektől. A rendszerek újszerű lehetőségei közül kiemelésre érdemes a szöveges, képi és hangformátumú dokumentumtárolás, több felhasználó egyidejű hozzáférése ugyanahhoz az adathoz, a kommentárhozzáfűzés (akár hang is lehet), valamint a hozzáférési szintek definiálhatósága.

A gyártó és a forgalmazó egyaránt úgy véli, hogy a rendszer sikerét nagymértékben segítheti a honosítás. Már készül a szoftver magyar nyelvű átirata, amely várhatóan 1-2 hónap alatt juthat el a felhasználókhoz.

Nem 6-ásvadászat!

Nem ártott volna „bespájzolni" néhány Magic-részvényt — vagy valamivel korábban egy jó adag Lotust —, s az ember mindjárt jobban érezné magát a dollárezreket számlálgatva... De míg a Lotus esetében az IBM bevásárlási szándéka verte fel az árakat, a Magicnek „elég volt" jó időben bejelentenie siker-fejlesztőrendszerének új változatát, s a New York-i tőzsdén máris 60 (!) százalékkal emelkedtek a cég részvényeinek árfolyamai.

A Magicet Magyarországon 1989 óta használják, elsősorban profi szoftverfejlesztők — igen széles alkalmazói körrel —, s a termék itthoni sikerét jól jelzi, hogy a hazai disztribútor Onyx a legjobb európai Magic-viszonteladó címet is kiérdemelte. A Magic 6-os 2. generációs kliens/szerver fejlesztőeszköz, amely egyidejű, többplatformos grafikus (GUI) és karakteres alkalmazásfejlesztést, rugalmas alkalmazásmegosztást, valamint a meglévő információk ellenőrzött továbbvitelét teszi lehetővé. Hatékony megoldásokat biztosít kódolás nélkül a fejlesztés, karbantartás és futtatás területén, heterogén környezetben, megőrizve a már meglévő információt, adatokat. Különösen olyan helyzetekben előnyös, amelyeknél a kliens/szerver szerkezet többplatformos környezetben működik, kevert grafikus és karakteres felületekkel. Az új változat a folyamatos elérhetőséget és a rugalmas kliens/szerver funkcionalitást megvalósítva sikeresen élhet együtt karakteres Magic szerverekkel.

A Magic 6 kihasználja a Windows GUI funkcionalitását, beleértve a méretezést, grafikát, 2- és 3-dimenziós árnyékolást, dialógusdobozokat, betűkészleteket, súgólehetőségeket, nyomtatást és még sok egyebet. Az új formátumszerkesztő a formátumok egyszerű és gyors szerkesztését a vázlatháló, a színbeállító skála, a méretezhető és áthelyezhető vezérlőfunkciók és drag and drop segítségével teszi lehetővé.

Támogatja az egyéb Windows-alkalmazásokkal való dinamikus adatcserét, a végfelhasználó munkáját teljesen integrált súgólehetőséggel és WinHelp-kompatibilitással segíti. A korábbi változatnál fejlettebb automatikus programgenerátort tartalmaz: automatikusan konvertálja a grafikus formátumokat karakteres képernyővé — így lehetővé teszi a grafikus és karakteres alkalmazások egyidejű fejlesztését, az egy alkalmazás fejlesztésével, illetve karbantartásával járó előnyöket kihasználva.

A Magic 6-os „lefelé kompatibilis" a korábbi változatokkal, a korábbi Magic-változatokban megvalósított alkalmazások egyszerűen áthozhatók az új környezetbe. Az alkalmazás régi és új változatai együtt is élhetnek, így a Windowsra való áttérés az igények szerint, az erőforrásoknak megfelelően történhet. A Magic 6 hivatalos világpremierje 1995. július 14-én volt a Magic Felhasználói Világkonferenciáján, a kaliforniai Palm Springsben. Az Onyxtól kapott információ szerint nyár végére elkészül a program magyarított változata is, alig egy hónappal követve az angol verzió megjelenését.

Digital — több kategóriában

A Digital — élve az Alpha processzor kínálta sebességelőnyökkel, valamint őrizve azt a több évtizedes hagyományt, mely szerint gépeik üzembiztonsága elsődleges követelmény — nem csak a csúcskategóriákban ígér és nyújt csúcsteljesítményt.

Csúcsgépeket nem szokás „angróban" vásárolni, még akkor sem, ha árban e gépek egyre közelebb jutnak a



megfizethető kategóriához. Márpedig az új Alpha gépek esetében csaknem ez történt. A felfokozott keresletet jellemzi, hogy már a bejelentéskor 70 gép „lábon” elkelt, s azóta jóval 200 fölé szökött ez a szám. Különösen népszerűek ezek a gépek a mobiltelefon-szolgáltatók körében: gyakorlatilag valós időben tudnak ellenőrizni minden hívást, s amennyiben a rendszer hamisított kártyát szúr ki, megakadályozható a telefonkapcsolat létrejötte.

Egy másik kategória új darabjának ára már leírva sem túl riasztó: a mintegy 500 000 forintért elérhető Prioris LX szerverek bizonyos technikai megoldásokban túl is mutatnak saját kategóriájukon (hőérzékelős hűtés, ECC-memóriatámogatás), kivételes megbízhatóságot nyújtva. Valódi plug and play szerverek, hiszen alaplapra integrálva tartalmazzák azokat a csatolókat (SCSI, Ethernet), amelyek nélkülözhetetlenek a szerver funkcióhoz. Processzorméretben az alsó határ a 486/66, a felső a 90 MHz-es Pentium, a memória maximuma 192 Mbájt, a cache-é 512 Kbájt. Kisebb munkacsoportok, vidéki irodák számára ajánlják elsősorban.

A törpekategóriában búcsút inthetünk a monokróm kijelzőnek, az induló merevlemezméret pedig 340 Mbájtra növekedett, mind a Hinote, mind Hinote Ultra notebookoknál. Az ergonómiai szempontok fokozott érvényesítése jelentős méret- és súlycsökkenést eredményezett mindkét család tagjainál. Minden kategóriára érvényes a Digitalnak az egész világra kiterjesztett, hároméves garanciája.

Rádiós adatátvitel

Mint ahogy a magyarországi telefonhálózat kiépítésében is jelentős szerep jut a rádiós átviteli megoldásoknak, ugyanúgy a számítógép-hálózatok „röghözkötöttségét” oldhatja az a technika, amelynek alkalmazásával a „madzag” (koax, sodrott érpár, optikai szál) szerepét a rádióhullámok veszik át.

Az Arlan (Advanced Radio LAN) család tagjai szórt spektrumú rádiófrekvenciás (SST) összeköttetési lehetőséget nyújtanak, részlegesen vagy egészében is kiépülhetnek segítségükkel a számítógépes hálózatok. A családtagok közül a 650-es egy teljesen vezeték nélküli LAN kommunikációs kártya, a 620-as „Bridge” két Ethernet hálózat vezeték nélküli összekötésére szolgál, a 610-es Ethernet hub pedig a 620-as kártyával felszerelt munkaállomásokkal teremt kapcsolatot. Az átviteli sebesség 1 Mbit/s, a frekvenciatartomány 2,445-2,475 GHz, a hatótávolság épületen belül 100 m, szabad téren 300 m, épületek között rálátással, irányított antennával pedig akár több km is lehet.

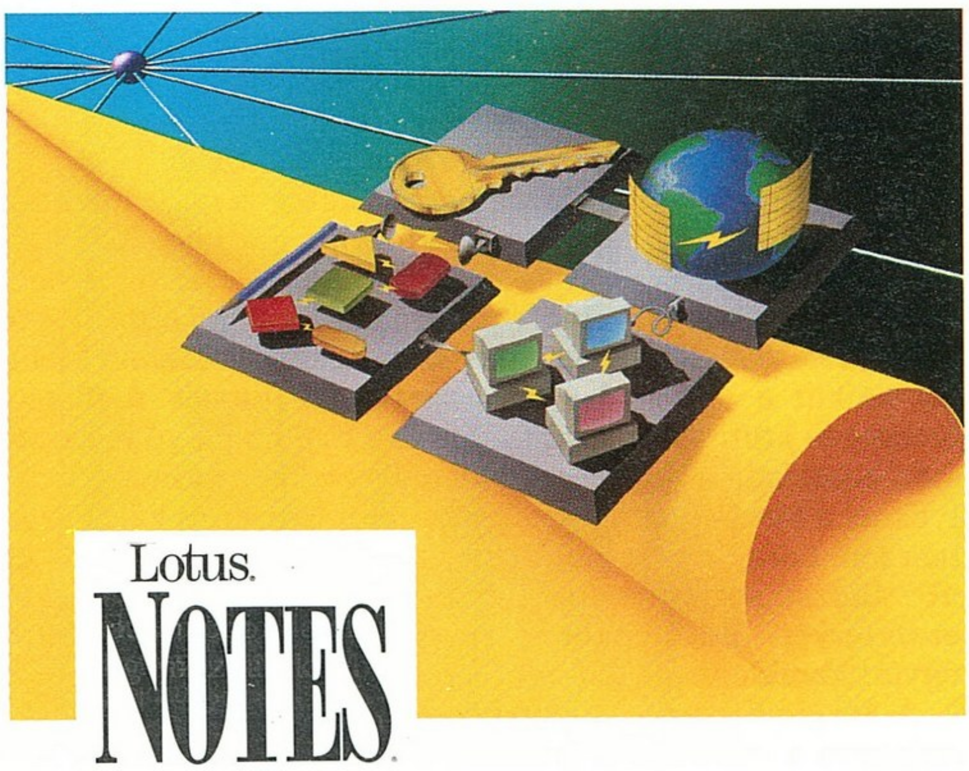
Az Arlan család alkalmazási előnye a rugalmas lehetőségekben van: meglévő hálózatok — akár átmeneti — bővítésére éppúgy szolgálhat, mint egy épületen belül elszórtan, több helyiségben működő gépek összeköttetésére, vagy akár épületek közötti adatátvitelre, hordozható számítógépbe építve pedig mozgás közben is a hálózat tagja marad a mobil egység.

A Twin Peaks Lotus-epizódja

Amikor nyár elején kiderült, hogy az IBM megvásárolja a Lotust, az érdeklődés rögtön a Notes programcsomagra terelődött. Feltehető volt ugyanis, hogy ez a hatékony eszköz az IBM stratégiájában bizonyosan komoly szerephez jut. A UniOffice Rendszerház véletlenül éppen akkoriban tartott egy egésznapos bemutatót a Lotus Notesról. Ennek kapcsán elmélyedhetnénk a szinkronizáció, a replikáció, a dokumentumkezelés, a titkosítás, a szövegkeresés, a GSM-kapcsolat és a többi szakmai téma rejtelmeiben, de ezt majd más alkalommal tesszük meg. Most érjék be a Lotus Notes jellegére szellemesen rávilágító egyik anekdotával, amelyet Suresh Patel, a Lotus kelet- és közép-európai marketingigazgatója mesélt el.

Annak idején a Magyarországon is bemutatott Twin Peaks amerikai sorozatot Japántól Argentináig a világ sok tévéállomása kisebb-nagyobb eltolódással közel egyszerre sugározta. Egyes Lotus-kirendeltségek munkatársai így még csak a hatodik résznél tartottak, mások a 14-ikinél, stb. Mindenki kíváncsi volt azonban a folytatásra, meg a kihagyott részekre, megindultak tehát a telefonhívások Londonból Tokióba, meg egyéb „szomszédos” országokba. Rohamosan nőtt a cég telefonszámlája.

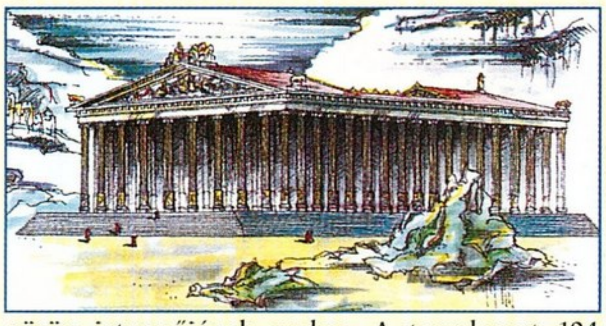
Ekkor a Lotus vezetői nem úgy reagáltak, hogy megtiltják és szigorúan ellenőrzik a telefonálgatást, hanem beállítottak egy Twin Peaks-szervert. A kliensek ugyebár adva voltak, a program pedig a Lotus Notes volt, azzal építették fel a Twin Peaks adatbázisát. Ehhez bárki hozzátehette saját kommentárjait, bevihetett szöveget, hanganyagot, videóra felvett részleteket, és üzengethettek is egymásnak a távoli kollégák. Gyorsabban és célravezetőbben lehetett a szerveren keresztül tájékozódni a Twin Peaks epizódjairól, történéseiről. És lássanak csodát: a napi többórás világméretű Twin Peaks-társalgás lecsökkent 15 perces kliens/szerver kommunikációra. A költségkímélésnek — és a reklámnak — ez kétségtelenül nagyon hatásos eszköze volt.



AZ ÓKORI VILÁG HÉT CSODÁJA

Az Artemisz-templom

A kisázsiai tengerparton, Kr. e. a VI. században épült szentélyben állt Artemisznek, a vadászat

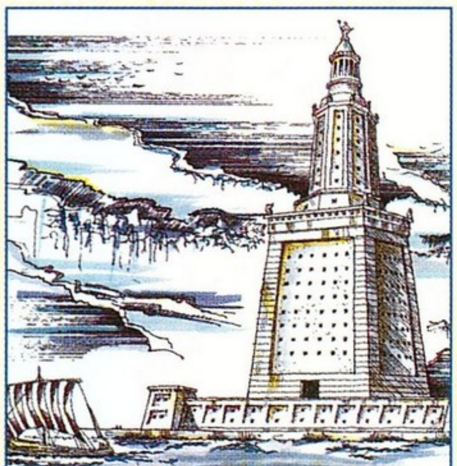


görög istennőjének szobra. A templomot 194 márványoszlop tartotta. A 18 méter magas oszlopok költségének túlnyomó részét a lüdiai király, Krőzus állta. Az **Info-Katalógus**-ban szereplő partnereink közül senkinek nem kell Krőzusnak lennie, mert hirdetéseink árai a Kr. u. II. évezred rossz gazdasági helyzete ellenére is mérsékeltek, sőt a MADE-INFO bérlet szolgáltatásunkkal akár másfél évig is garantáltak.

...ráadásul Krőzus még az áfát sem igényelhette vissza.

Az alexandriai világítótorony

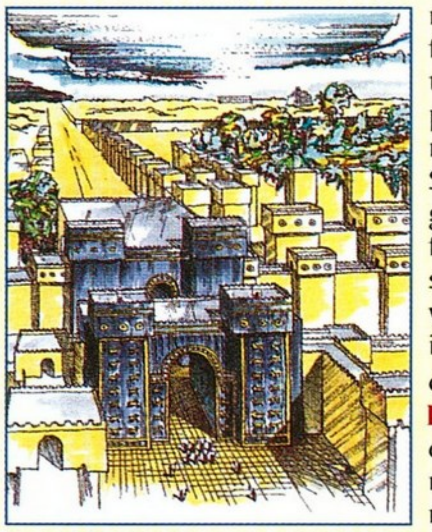
A világítótorony a Kr. e. III. században épült Pharosz szigetén és a XIV. század elejéig tündökölt. A



világon ez volt az első olyan ismert építmény, amely a hajók irányítására, az emberek tájékozódására szolgált. Már kilencedik éve hasonló szerepet tölt be az **Info-Katalógus** az információtechnikai piacon. A kiadvány tematikus tárgymutatója a céltudatos keresés iránytűje, valamint szakmai partnerkereső fejezetének eredménye számos üzleti kapcsolat létrejötte.

Szemiramisz függőkertje

Kr. e. a VI. században, Babilonban a sivatag felől érkező utazót elkápráztatta a királyi palota körül ékeskedő

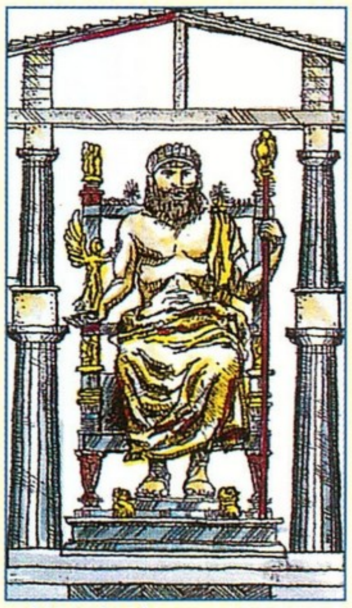


függőkertek teraszainak pompás zöld növényzete. Sokszínűségével az informatika száraz, szürke világában ilyen üde oázis az **Info-Katalógus**. Az egységes formátum ellenére tág teret kapnak a kitűnni vágyók is: a színes borítókkal, az elválasztó kartonokkal, a poszterrel, a könyvjelzővel.

...és a Pantone színskála szinte végtelen!

Az olümpiai Zeusz-szobor

Olümpiában, Kr. e. az V. században Pheidiász, a híres szobrász megalkotta Zeusz hatalmas,



arany-elefántcsont borítású, mitológiai jelenetekkel díszített trónuson ülő szobrát. Ahogy Zeusz, mint főisten rendet tartott a görög istenek világában, úgy igazítanak el az **Info-Katalógus**-ban a kötetenkénti cégmutatók és a szakmai területen a tematikus tárgymutatók.

...kérjük, hogy „INFOSZ", az információtechnika istene adja áldását partnereink további sikeres tevékenységéhez.

Mauszolosz sírja

Kisázsiában, Mauszolosz a Kr. e. IV. sz.-ban építtette a görög építőművészet első kétemeletes síremlékét,



amely olyan pompás volt, hogy ma is így hívják az impozáns halotti emlékműveket: mauzóleum. A 49 m magas épületegyüttes négy részből állt; a színes domborművekkel díszített alapzatból, a 36 oszlopos főépületből, a 24 lépcsős tetőpiramisból, és az ún. quadrigából (négylovas szekér). Ugyanilyen négy önálló egységet képez az **Info-Katalógus** négy kötete is egy kiadványban: HARDVER hálózat, ups, elektronika, automatizálás, SZOFTVER térinformatika, vonalkódtechnika, IRODATECHNIKA irodaszer, irodabútor, nyomdatechnika, papíripar, TÁVKÖZLÉS biztonságtechnika, tűzvédelem.

...ahogy Mauszolosz síremléke is egy örök emlék, úgy az Ön **Info-Katalógus**-ban megjelenő hirdetése is időt álló.

A rhodoszi kolosszus

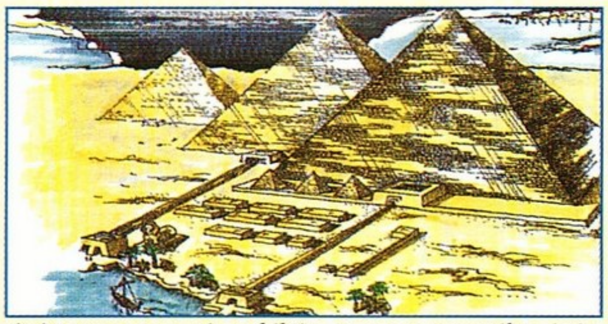
Rhodosz szigetén, Kr. e. 290-ben építették Héliosz óriás méretű bronzszobrát, amely kb. 35



m-es magasságával messziről mutatta a tengerészeknek a sziget kereskedelmi kikötőjének bejáratát. Ahogy az ókori görög világban a Kolosszus egy biztos pont volt a kereskedők tájékozódásában, úgy napjainkban az eladni és vásárolni szándékozóknak az **Info-Katalógus** is az, számos kereskedelmi szolgáltatásával: a floppylemezen megjelenő szakmai telefonkönyvvel, az egyedülálló terjesztésével, az üzleti partnerkereső fejezeteivel, az információkérő levelezőlapokkal és a vásárlási kedvezményekre jogosító bónuszrovatával.

Kheopsz piramisa

Kheopsz, az ókori Egyiptom IV. dinasztiájának uralkodója Kr. e. 2520 körül százezer egyiptomival



építtette meg síremlékét. A monumentális építmény 230 méter oldalszélességű, 137 méter magas és 2 300 000 kőtömbből áll. Az ókori világ csodái közül ez az egyetlen, amely napjainkig fennmaradt. Hasonlóan az **Info-Katalógus** az egyetlen, amely a szakmai elismerés eredményeként már kilencedik éve rendszeresen megjelenik. Félévről félévre nő a benne szereplő cégek száma, és egyre több szolgáltatás kapcsolódik hozzá.

...és nem utolsósorban, közel annyi betűből áll, ahány kőtömbből a Kheopsz piramis.

Info-Katalógus '95 II.

lemezmelléklettel

Kiadó	: **MADE-INFO KFT.**
Levélcím	: 1506 Budapest, Pf. 99
Telefon	: 227-3647
Telefax	: 228-1934

A TÁJÉKOZOTTAK KÉZIKÖNYVE

Előfizetünk az **Info-Katalógus '95 II.** félévi számára **1200 Ft**-os áfás áron, melynek összegét a mai napon átutaltuk a MADE-INFO KFT. **Postabank 219-98636/021-16565** számú számlájára! (A Kiadó a pénz beérkezése után azonnal számlát küld, és a kiadványt megjelenés után postázza.)

Cégnév: ....

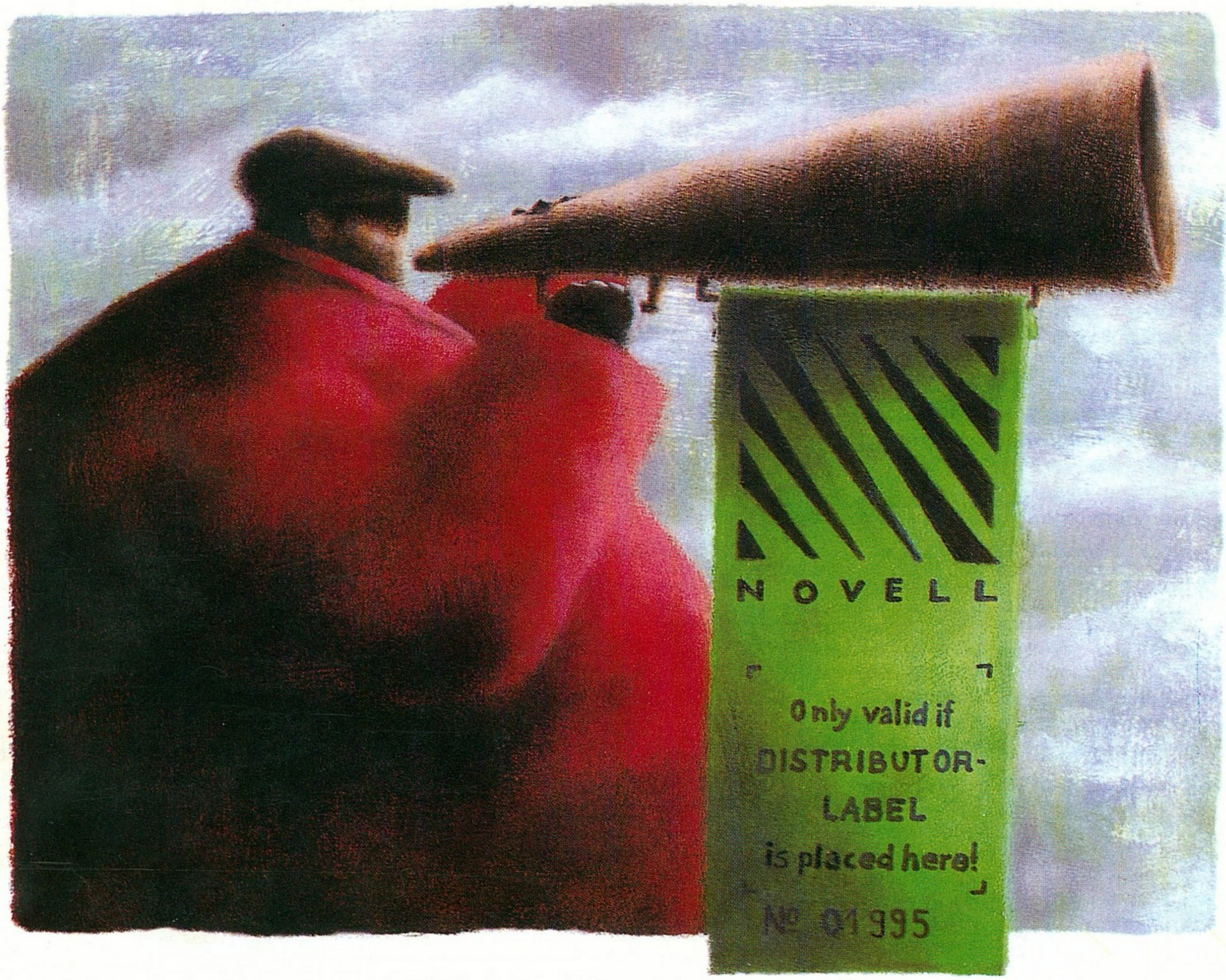
Ügyintéző: ....

Pontos cím: ....

....

Telefon: ....

Zöld jelzés a Novell-címkéknek



Ha az Ön által vásárolt Novell-termékről hiányzik a sorozatszámmal ellátott, zöld címke, lehet, hogy másvalamit is nélkülöznie kell – a támogatást.

Abban az esetben, ha a dobozon rajta van az itt látható címke, minden rendben.

Ez azt jelzi, hogy a termék egy feljogosított Novell disztribútortól származik. A címke nélkül viszont ezt egyszerűen nem lehet elismerni, és így viszonteladója számára semmiféle értékesítési támogatást nem biztosítanak.

Nyilvánvalóan nem szeretne támogatás nélkül maradni, ezért kizárólag hivatalos Novell-termékeket érdemes vásárolnia.

Ha kételkedik a termék eredetiségében, egyszerűen csak lépjen kapcsolatba az alább felsorolt feljogosított Novell disztribútorok valamelyikével !

NOVELL

The Past, Present, and Future of Network Computing.

3Soft Számítástechnikai Kft.
1123 Budapest, Kapitány u. 6.
Tel.: 212-2552 Fax: 156-5419

Computer 2000 Magyarország Kft.
1027 Budapest, Kapás u. 11-15.
Tel.: 202-4520 Fax: 202-4529

Walton Networking Kft.
1077 Budapest, Almássy tér 2.
Tel.: 267-9010 Fax: 267-9011

INFORMÁCIÓKÉRÉS: 0821